

Berufssakademie Sachsen

Staatliche Studienakademie Leipzig

Evaluierung der SAP Cloud Plattform für die Entwicklung
und Anwendung (energiewirtschaftlicher) Funktionen (Apps)
am Beispiel einer selbst entwickelten Funktion

Bachelorarbeit

zur Erlangung des akademischen Grades eines

„Bachelor of Science“

in der Studienrichtung Informatik

Eingereicht von: Angela Stöckert

Herrmann-Meyer-Straße 11, 04207 Leipzig

Seminargruppe: CS14-1 /CS15-1

Matrikelnr.: 5000559

Betreuer: M. Sc. André Kierzkowski

Arvato Systems Perdata GmbH

Martin-Luther-Ring 7-9

04109 Leipzig

Leipzig, 26.06.2018

Inhaltsverzeichnis

1 Einführung in die Thematik	1
1.1 Vorwort.....	1
1.2 Motivation und Zielstellung	1
1.3 Thesen	3
2 Vorstellung der SAP Cloud Platform	4
2.1 Plattform.....	4
2.2 Cloud-Computing.....	4
2.3 SAP Cloud Platform.....	8
2.3.1 Zielgruppe	9
2.3.2 Umgebungen und Regionen.....	9
2.3.3 Dienste	12
2.3.4 Zugang zu Diensten auf der SAP Cloud Plattform und Bezahlung.....	14
2.3.5 Anbindung an andere Systeme	16
2.3.6 Verfügbarkeit und Wartung.....	17
3 Vorüberlegungen zur Implementierung einer Webanwendung	19
3.1 Bestehende Anwendung mit Eigenentwicklung	19
3.2 Anforderungen an die Anwendung	20
3.3 Fiori	21
3.4 Verfügbarkeit und Erweiterung von Standardfunktionen	23
3.5 Architektur der eigenen Anwendung	24
3.6 Sicherheit und Datenschutz.....	26
3.7 Entwicklungsumgebung.....	30
3.8 Evaluierungskriterien	32
4 Evaluierung der SAP Cloud Platform	34
4.1 Umsetzung der Webanwendung.....	34
4.1.1 Verbindung zwischen SAP Backend und SAP Cloud Platform	35
4.1.2 Einrichten eines OData-Dienstes (dessen Nutzung)	39
4.1.3 Erstellen und Anpassen der Oberfläche	42
4.1.4 Verteilung (Deployment) der Anwendung	48
4.1.5 Erreichbarkeit der Anwendung für Nutzer	51
4.1.6 Verwendung der Anwendung mit einem Standard-OData-Dienst	52
4.2 Einschätzung der SAP Cloud Platform	53
4.2.1 Entwicklungsumgebung SAP Web IDE.....	53
4.2.2 Verbindungen zur Cloud Platform.....	54
4.2.3 SAP Cloud Plattform als Ganzes	55

5 Fazit und Ausblick.....	58
5.1 Probleme	59
5.2 Ausblick.....	60
6 Abkürzungsverzeichnis	IV
7 Abbildungsverzeichnis	VI
8 Tabellenverzeichnis	VII
9 Literaturverzeichnis	VIII
10 Anhangsverzeichnis	XVIII
11 Anhang.....	XIX
12 Selbstständigkeitserklärung	XLIII

1 Einführung in die Thematik

1.1 Vorwort

- Funktionen
- Systemnamen
- Quellen []
- Fußnote

Dieses Kapitel beschreibt die Zielstellung sowie den Hintergrund dieser Arbeit und stellt das Unternehmen, für welches diese Arbeit erstellt wird, kurz vor. Anschließend listet es die Arbeitsthesen auf.

Um Funktionen im Sinne von Fähigkeiten von Funktionen der Programmierung abzugrenzen sollen Erstere innerhalb dieser Arbeit vorwiegend als *Anwendung* oder *Applikation* (kurz App) bezeichnet werden.

Die in dieser Arbeit verwendet Systemnamen SGW (kurz für SAP Gateway) und SBE (kurz für SAP Backend) wurden aus Datenschutzgründen verändert und existieren so nicht in der Realität.

1.2 Motivation und Zielstellung

Im Rahmen dieser Arbeit gilt es zu evaluieren, ob die SAP Cloud Platform in der Lage ist, energiewirtschaftliche Individualprogrammierungen aus dem System zur Planung von Unternehmensressourcen *SAP ERP Central Component 6.0* (ECC), als *SAP-Fiori*-Anwendungen abzubilden. Hierfür sei ein Beispiel auszuwählen und umzusetzen und daran zu prüfen, ob dessen Oberfläche nach einmaliger Implementierung auch für Energieversorger mit Standardausprägung gleichermaßen nutzbar ist. In einem solchen Fall erwartet das Unternehmen *Arvato Systems Perdata GmbH* (ASP) zukünftig eine drastische Verringerung der Wartungskosten für die Eigenentwicklung. Untersucht wird auch, ob durch die Verwendung von Diensten in der SAP Cloud Platform zur Entwicklung des Beispiels ein Mehrwert entsteht.

Die Arvato Systems GmbH ist Teil der Arvato AG und als solche auch Teil des *Bertelsmann* Konzerns mit Hauptsitz in Gütersloh. Sie konzentriert sich auf Dienstleistungen im Informationstechnik-Sektor (IT-Sektor). Die Geschäftseinheit ASP, unter anderem am Standort Leipzig, hat sich insbesondere auf Dienstleistungen für Unternehmen der Ver- und Entsorgungswirtschaft spezialisiert. Dabei entwickelt ASP mit rund 300 Mitarbeitern individuelle IT-Lösungen, welche exakt auf die Bedürfnisse des Kunden abgestimmt sind. Die Kernkompetenz liegt besonders bei der Planung, Bereitstellung und im Betrieb von SAP-basierten Systemen.

Ein Kunde, welcher zu ASP kommt, hat in der Regel bereits bestehende, historisch gewachsene Geschäftsprozesse, die das System zur Ressourcenplanung eines Unternehmens (engl. Enterprise Resource Planning (ERP)) der SAP SE¹ (SAP) in der Standardversion oftmals nur unzureichend abbilden kann. Anpassungen der ausgelieferten Funktionalitäten an die spezifischen betriebswirtschaftlichen Anforderungen des Kunden (*Customizing*) ist im ERP durch Parametrisierung vorgesehen [1 S. 11]. „Der Anwender kann so den Funktionsumfang sowie den Programmablauf der Standardsoftware steuern und gemäß seinen Vorstellungen modifizieren [2].“ Für die Erweiterung von Tabellen, welche auch in der Businesslogik beachtet werden, stehen *Customizing Includes* (CI) zur Verfügung.

Die Möglichkeit der Erweiterungsprogrammierung wird nur dann ergriffen, wenn der durch das Customizing vorgegebene Rahmen nicht ausreicht [3], da sie mit einer Einschränkung der Garantie für die Kompatibilität von Aktualisierungen durch den Hersteller einhergeht [4 S. 338]. Garantien betrachten die Standardsoftware und berücksichtigen nur jene Schnittstellen zu Individualprogrammierungen, welche in den von SAP vorgesehenen Platzhaltern untergebracht sind. Diese sind durch vordefinierte Schnittstellen mit dem ERP verbunden. Individuelle Programmbestandteile außerhalb dieser Platzhalter müssen nach dem Einspielen einer neuen Programmversion manuell auf ihre Funktionsfähigkeit geprüft und bestehende Probleme behoben werden. Dies ist für ASP mit Zeit und für den Kunden mit Kosten verbunden. Dennoch gibt es Fälle, in denen die Individualisierungen des Kunden so wichtig sind, dass er ASP mit Erweiterungsprogrammierungen außerhalb von Platzhaltern beauftragt.

Aber nicht nur bei Release-, sondern auch bei Produktwechseln kann es zu hohen Aufwänden in Bezug zu Erweiterungsprogrammierungen kommen. SAP SE gab im Oktober 2014 bekannt, dass die aktuelle ERP-Produktlinie *SAP ERP Central Component 6.0* (ECC) (oft ebenso als *Business Suite* bezeichnet) ab 2026 nicht mehr unterstützt wird [5]. Kunden der ASP benötigen daher in den nächsten Jahren ein Folgesystem, welches ihre Geschäftsprozesse abbilden kann und ihren Anforderungen an Stabilität, Zukunftsfähigkeit und Sicherheit entspricht. ASP sieht in der *SAP Business Suite 4 HANA* (SAP S/4 HANA) eine solche Alternative. Das System unterscheidet sich vom vorherigen ERP in vielen Punkten. Während bei ECC die Wahl der Datenbank dem Kunden überlassen ist, benutzt S/4 HANA zwingend *SAP HANA*, eine *In-Memory*-Platform, welche Datenzugriffe beschleunigen und Auswertungen in Echtzeit ermöglichen soll.

¹ Societas Europaea (SE) „ist eine Rechtsform für Aktiengesellschaften in der Europäischen Union und im Europäischen Wirtschaftsraum“ [55].

Zudem bietet S/4 HANA neue Benutzeroberflächen, welche unter dem Namen *SAP Fiori* vorgestellt wurden [6]. Diese machen einzelne betriebswirtschaftliche Anforderungen unter Berücksichtigung der Rolle und der Berechtigungen eines Nutzers als Apps auf allen Endgeräten verfügbar [7]. Solche modernen Benutzeroberflächen können bereits für ECC-Systeme eingerichtet werden, um den Kunden einen fließenden Übergang zum neuen System zu ermöglichen. Zum Beispiel ist es denkbar, im ersten Schritt einzelne Programme im ERP-System durch Webapplikationen zu ersetzen [8 S. 20], damit Nutzer sich an die neuen Oberflächen und Navigationswege gewöhnen können. Passende Werkzeuge dafür bietet SAP auf der *SAP Cloud Platform* (früher *SAP HANA Cloud Platform*) an [9]. Bei ihr handelt es sich um eine Entwicklungsplattform mit deren Hilfsmitteln unter anderem Fiori-Oberflächen erweitert und angepasst werden [10] können.

ASP steht aktuell vor der Aufgabe, Ihren Kunden, den Energieversorgern, den Umstieg auf moderne SAP-Systeme, wie S/4 HANA zu ermöglichen und dabei die Kosten, so gering wie möglich zu halten. Daher ist es für ASP von großer Bedeutung zu prüfen, ob eine Eigenentwicklung aus einem ECC-System durch eine Fiori-Anwendung abgebildet werden kann.

1.3 Thesen

Folgende Thesen gilt es in der Analyse näher zu untersuchen.

- Es ist mit der Cloud Platform möglich, Fiori-Anwendungen zu entwickeln, die ihre Daten aus ECC-Systemen beziehen.
- Für die Anpassung und Erweiterung von Fiori-Benutzeroberflächen ist die Cloud Platform unverzichtbar.
- Individualprogrammierungen für Kunden können mit den Werkzeugen der SAP Cloud Platform als Fiori-Apps umgesetzt werden.
- Fiori-Apps, welche eine Eigenentwicklung eines Kunden abbilden, können ohne Änderungen des Quelltextes auch für andere Kunden mit Standardausprägung genutzt werden.
- Die SAP Cloud Platform stellt einen Mehrwert bei der Entwicklung von Webanwendungen dar.

2 Vorstellung der SAP Cloud Platform

In diesem Kapitel sollen die namensgebenden Begriffe Plattform und Cloud-Computing definiert und die SAP Cloud Platform vorgestellt werden.

2.1 Plattform

Der Begriff Plattform ist nicht IT-spezifisch, hinter ihm verbirgt sich jedoch weit mehr als nur eine *erhöhte Ebene*, wie im allgemeinsprachlichen Sinn. Er existiert in verschiedenen Anwendungsdomänen, wie zum Beispiel der Automobilindustrie, wo man darunter eine technische Basis versteht, die für verschiedene Fahrzeugmodelle gleich verwendet wird. So bauen zum Beispiel der *VW Polo*, der *Seat Ibiza* und der *Skoda Fabia* auf derselben Plattform, bestehend aus Bodenplatte, Tank, Auspuffanlage, Heizung und weiteren Teilen auf [11].

In der Informatik finden sich unzählige Plattformen, die sich zuerst einmal in Hardware- und Software-Plattformen einteilen lassen. Erstere bestehen aus Rechnerarchitektur mit ihren Arbeits-, Befehls- und Prozessstrukturen. Beispiele für Minicomputer-Plattformen sind *Arduino* oder *Raspberry Pi*. Software-Plattformen können mehrschichtig existieren. Die untere Schicht, zumeist ein Betriebssystem, läuft auf einer Hardware-Plattform und kann wiederum Basis für weitere Software-Plattformen sein, wie zum Beispiel für das Internet. Das Internet selbst ist die Basis für verschiedene Arten von Plattformen, wie zum Beispiel Suchmaschinen (*Google*), Vergleichsportale (*Check24*), Marktplätze (*Amazon*), Sharing-Economy-Plattformen (*AirBnB*, *Mitfahrzentrale*), Inhaltsdienste (*Youtube*) und soziale Netzwerke (*Facebook*) [12 S. 4].

Für Softwareentwickler besonders interessant sind so genannte Entwicklungsplattformen. Dies sind Plattformen, die Schnittstellen zur Infrastruktur und Hilfsmittel zur Entwicklung von Anwendungen zur Verfügung stellen [13 S. 8]. Die Schnittstellen sind in der Regel standardisiert [14 S. 95]. Plattformen ermöglichen dem Entwickler, eigene Dienste zu definieren, die kompatibel mit denen von Partnern, Konkurrenten und Kunden sind, welche die gleiche Plattform nutzen [15 S. 625].

2.2 Cloud-Computing

In der Literatur finden sich unterschiedlichste Versionen von Definitionen des Cloud-Computing-Begriffs. 2011 hat das *National Institute of Standards and Technology* (NIST) zur Entwirrung beigetragen und eine offizielle Definition von Cloud-Computing publiziert [16], welche Charakteristika, Service-Modelle und Verteilungsmodelle festlegt. Auch in der deutschsprachigen Literatur finden sich mehrere Bezüge auf diese Definition [17 S. 5], [14]

S. 95], [18 S. 69], daher soll sie auch in dieser Arbeit Anwendung finden. Das NIST definiert fünf wesentliche Merkmale für Cloud-Computing:

- Der Zugriff des Nutzers auf die ihm zur Verfügung gestellten Dienste erfolgt nach Bedarf selbstständig und ohne menschliche Interaktion.
- Die Dienste sind auf verschiedenen Endgeräten (PC, Laptop, Server, Smartphones und Tablets) erreichbar.
- Die Dienste des Anbieters sind in einem Ressourcenvorrat gesammelt und werden von mehreren Nutzern zeitgleich unter Berücksichtigung derer Bedarfe genutzt.
- Die Dienste können schnell und in einigen Fällen sogar automatisch gebucht und freigegeben werden um sich „elastisch“ dem Bedarf des Nutzers anzupassen. Der Nutzer hat dabei den Eindruck, dass ihm unbegrenzte Ressourcen zur Verfügung stehen.
- Die Dienste können automatisch kontrolliert und der Ressourcenverbrauch optimiert werden. Diese Messung der Dienste mit Hilfe geeigneter Ressourcen sorgt für Transparenz für den Anbieter und den Nutzer der Cloud.

Die in der Cloud angebotenen Dienste werden allgemein in drei Serviceebenen unterteilt: *Infrastructure-as-a-Service* (IaaS), *Platform-as-a-Service* (PaaS) und *Software-as-a-Service* (SaaS). Baun [17 S. 39] erwähnt eine vierte Ebene, *Humans-as-a-Service* (HuaaS). Wie in Abbildung 1 erkennbar, unterscheiden sich die Serviceebenen darin, welcher Teil der IT vom Dienstleister betreut wird und welches Fachwissen der Kunde selbst braucht, um Angebote auf dieser Ebene zu nutzen. Bei IaaS wird dem Nutzer, zum Beispiel der IT-Abteilung einer Firma, eine abstrahierte Sicht auf die vom Cloud-Anbieter zur Verfügung gestellte Hardware geboten. Er kann diese Nutzen um Betriebssysteme zu installieren, Netzwerktopologien zu definieren oder um beanspruchte Kapazitäten zu skalieren [17 S. 32]. Große Anbieter in dieser Kategorie sind Amazon Web Services, Microsoft und Google. Die Berechnung der Kosten auf dieser Ebene gestaltet sich komplex, da Kosten für Rechenzeit und –leistung, für Datenbanken und –speicher, für Anfragen zwischen Client und Server und für übertragene Daten anfallen können [19 S. 47].

Dienste der PaaS-Ebene richten sich typischerweise an Entwickler oder IT-Architekten und beinhalten Entwicklungs- und Laufzeitumgebungen [17 S. 35], welche die Zeit bis zur Fertigstellung der Anwendung deutlich verkürzen können, da sich der Nutzer nicht zusätzlich um die zugrunde liegende Hardware kümmern muss [19 S. 44]. Auf dieser Ebene entstehen Entwicklungskosten, die nach Anzahl der Entwickler pro Monat berechnet werden und zusätzlich Entwicklerlizenzen enthalten können. Eventuell muss auch der Transfer von Daten bei der Berechnung beachtet werden [19 S. 47].

Auf der obersten Ebene können Endnutzer von SaaS profitieren. Sie nutzen eine Anwendung von jedem Cloud fähigen Endgerät aus an jedem Ort und müssen nicht über die Ressourcen nachdenken, welche die Anwendung benötigt. Dafür müssen Sie jedoch auch in manchen Fällen Einschränkungen in der Konfigurierbarkeit der Software akzeptieren [16 S. 6]. Ein populäres Beispiel hierfür ist das Cloud-Gaming, welches dem Nutzer ermöglicht ein Spiel, welches hohe Anforderungen an die Grafikkarte eines PCs stellt, auf einem Smartphone oder Tablet mit schwächerer Grafikkarte zu spielen [17 S. 70]. Die Abrechnung der Services erfolgt meist pro Nutzer in Form einer monatlichen Gebühr [19 S. 47].

Die Cloud-Dienste können in unterschiedlichen Cloud-Typen angeboten werden. Eine öffentliche Cloud (*public cloud*) wird von einem externen Dienstleister angeboten und betrieben [19 S. 45]. Der Nutzer kann meist mit Hilfe eines Webportals selbstständig den benötigten Leistungsumfang angeben [17 S. 27-28]. Manchmal findet sich hier in der Literatur auch der Begriff *external cloud*.

Bei einer privaten Cloud (*private cloud*, manchmal auch *internal cloud*) gibt es in der Literatur verschiedene Definition. Laut Baun [17] gehören der Anbieter und der Benutzer der Cloud zur selben organisatorischen Einheit, demselben Unternehmen. In manchen Fällen können auch Kunden oder Partner des Nutzers zugreifen. Das NIST legt das Hauptaugenmerk der Kategorisierung darauf, dass die Cloud-Infrastruktur bei einer privaten Cloud für die ausschließliche Nutzung durch eine Organisation, eventuell mit mehreren Abteilungen, gedacht ist. Laut NIST spielt es dabei keine Rolle, wo die Infrastruktur sich befindet, wer sie besitzt und wer sie betreibt.

Barton [19 S. 46] beschreibt Unterformen der privaten Cloud, welche von genau diesen Faktoren abhängen. Eine private Cloud, welche der Nutzer selbst im eigenen Rechenzentrum betreibt, nennt er *insourced private cloud*. Wird der Betrieb an externe Dienstleister übergeben werden, hängt seine Definition davon ab, ob die Infrastruktur dem externen Dienstleister auch gehört und physisch bei ihm befindet (*outsourced*), oder ob er auf Infrastruktur des Kunden zurückgreift (*managed*). Der Nutzer bleibt bei allen Formen von privaten Clouds Besitzer seiner Daten und hat die volle Kontrolle über die Einhaltung von Datenschutzverordnungen in Bezug auf personenbezogene Daten [17 S. 27-28]. Werden Dienste aus privaten und öffentlichen Clouds zusammengesetzt, spricht man von einer hybriden Cloud. Bei Lastspitzen oder bestimmten, eigenständigen Funktionen kann es sinnvoll sein, diese in eine öffentliche Cloud auszulagern, während der Rest des Systems in einer privaten Cloud verwaltet wird. Hierbei ist es wichtig zu prüfen, ob die Auslagerung in Konflikt mit den Sicherheitsbestimmungen des Nutzers liegt. Personenbezogene Daten sollten auch bei

hybrider Cloudnutzung im privaten Systemteil verbleiben [17 S. 29]. Ebenfalls spricht man von einer hybriden Cloud, wenn eine Cloud mit einer traditionellen IT-Landschaft verbunden wird [19 S. 46].

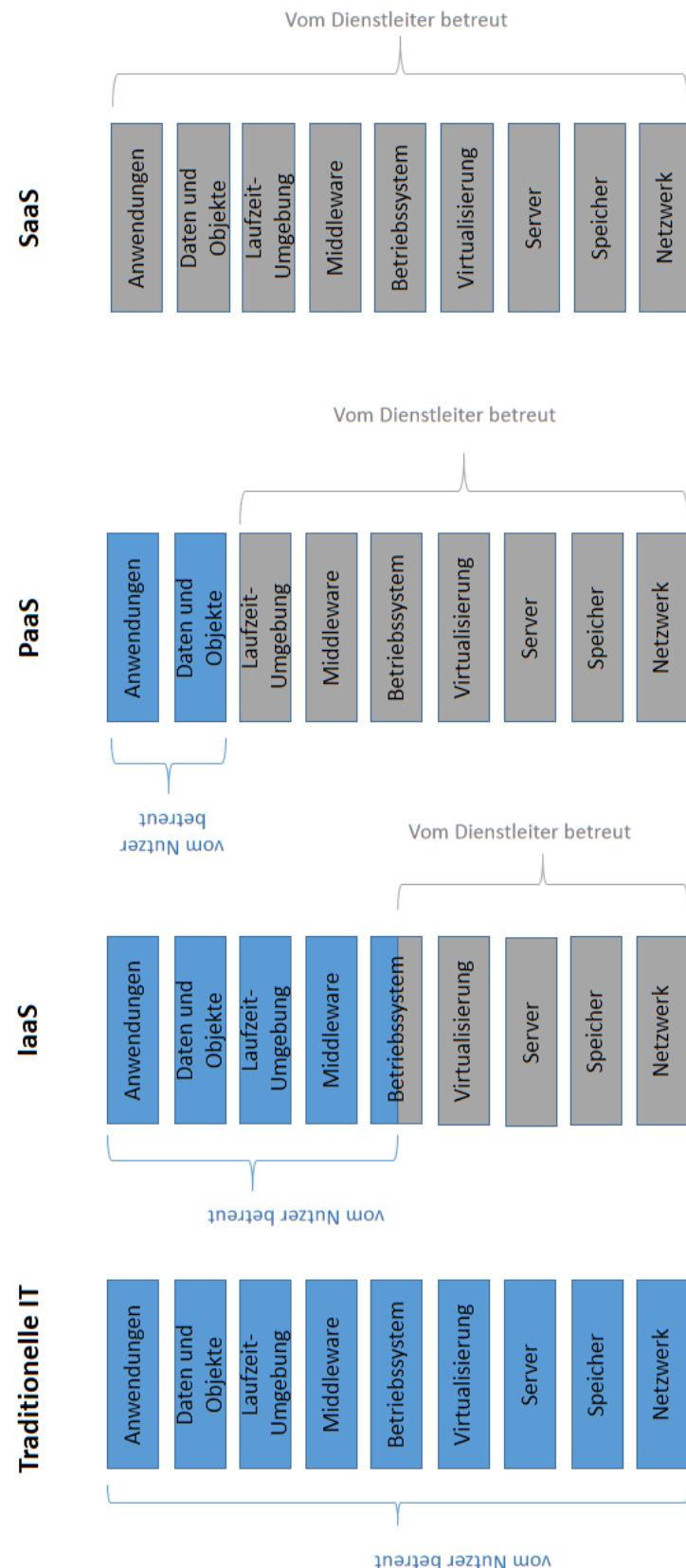


Abbildung 1: Serviceebenen in der Cloud [20]

2.3 SAP Cloud Platform

Bei der SAP Cloud Platform handelt sich um ein Platform-as-a-Service-Angebot, welches umfassende Dienste und Fähigkeiten der Applikationsentwicklung bereitstellt, die es dem Nutzer ermöglichen Geschäftsanwendungen in der Cloud zu bauen, zu erweitern und mit jeder modernen SAP-Software [21 S. 126] sowie mit Software anderer Anbieter zu integrieren [22 S. 8].

**Abbildung 2: SAP HANA Cloud Platform [23]**

Ihren Beginn fand die SAP Cloud Platform im Oktober 2012 als **SAP HANA Cloud** mit den Hauptbestandteilen **SAP HANA AppServices** und **SAP HANA DBServices**, wie Abbildung 2 zeigt. SAP HANA DBServices bezeichnet die SAP-HANA-Datenbank selbst und Hilfsmittel, mit welchen die Daten durchsucht oder bearbeitet werden können um anschließend als Grundlage für Vorhersagen zu dienen. Hier finden sich spezielle Suchalgorithmen, Strukturerkennungsalgorithmen, raumbezogene Analysen oder Diagramme.

Darauf aufbauend findet man zwei Arten von **SAP AppServices**: die **Enablement Services** und die **Application Services**. Erstere stellen Dienste zur Verwaltung von Applikationen und Systemen Dienste zur Verfügung, welche zum Beispiel Persistenz, Konnektivität, sowie

Identitäts- und Dokumentenverwaltung ermöglichen. Letztere enthalten Werkzeuge für spezielle Szenarien, zum Beispiel für mobile Applikationen, Portale, für die Zusammenarbeit innerhalb der Firma und mit Partnern sowie zur Integration.

In den folgenden Jahren hat SAP die Platform erweitert und ausgebaut und sie schließlich im Februar 2017 neu unter dem Namen *SAP Cloud Platform* veröffentlicht. Im April 2018 stehen dem Nutzer auf der SAP Cloud Platform Dienste zwölf verschiedener Kategorien zur Verfügung (Abbildung 5): *Analytics, Business Services, Collaboration, Data & Storage, DevOps, Integration, Internet of Things, Machine Learning, Mobile Services, Runtimes & Containers, Security und User Experience*. Die Services werden in Abschnitt 2.3.3 detailliert behandelt.

2.3.1 Zielgruppe

Nach Aussage von Matthias Steiner, einem Technik-Experte von SAP, bietet die SAP Cloud Plattform einen Mehrwert für neugegründete Firmen, welche eine Applikation auf Basis von SAP HANA bauen wollen, für Partner, die eine Erweiterung anbieten möchten, oder für Kunden, denen eine speziell angepasste Lösung für die von ihnen genutzten SAP-Anwendungen fehlt [23]. Doch nicht nur neugegründete Firmen, sondern auch die Firmen *Bosch* und *Siemens*, die *Hamburger Hafenbehörde* und der Gabelstaplerhersteller *Still GmbH* haben bereits Projekte mit der SAP Cloud Plattform umgesetzt. Siemens ist dadurch zu Vorhersagen über anfallende Wartungen in der Lage, Bosch ermöglicht LKW-Fahrern per App einen sicheren Stellplatz für die Nacht zu finden, im Hamburger Hafen werden in Echtzeit Containerschiffe koordiniert und bei Still GmbH Gabelstapler zu autonomem Handeln befähigt [24].

Partner von SAP nutzen die Fähigkeiten der Cloud Platform zur Erstellung von Erweiterungen bestehender SAP-Systeme und verkaufen diese dann im *SAP App Center*. Bestehende SAP-Kunden suchen dort nach einer passenden Lösung für ihr Problem. Wenn sie nicht fündig werden, oder nicht bereit sind einer anderen Firma den gewünschten Betrag für eine Anwendung zu zahlen, können sie die SAP Cloud Platform nutzen um SAP-Systeme nach ihren Wünschen anzupassen und zu erweitern.

2.3.2 Umgebungen und Regionen

Meldet sich ein Nutzer an der Testversion der SAP Cloud Platform an, empfängt ihn eine Startseite, welche bereits einige nützliche Navigationsziele (siehe Abbildung 3) enthält. Links oben befindet sich ein Menü, welches von der gewählten Umgebung abhängig ist. Links unten findet sich eine Verknüpfung zu nützlichen Internetadressen (*Useful Links*). Dort befindet sich unter anderem die Dokumentation der Entwicklungsplattform. Rechts oben fin-

det der Nutzer Informationen zu seinem Nutzerkonto. Dort kann er sich für eine der angebotenen Sprachen Englisch, vereinfachtes Chinesisch, Koreanisch oder Japanisch entscheiden.

Die SAP Cloud Platform bietet Entwicklern zwei Umgebungen für ihre Entwicklungen an. Die *Cloud-Foundry*- und die *Neo*-Umgebung. Cloud Foundry ist ein quelloffenes PaaS zur Entwicklung von Anwendungen, welches sich durch die unternehmensfähige Architektur, Portabilität, Multi-Cloud-Funktionen und offenen Standardprotokollen von den Konkurrenten abhebt [25]. Unter *Multi-Cloud* versteht man, dass die Plattform auf Infrastrukturdiensten von unterschiedlichen Anbietern betrieben wird [26].

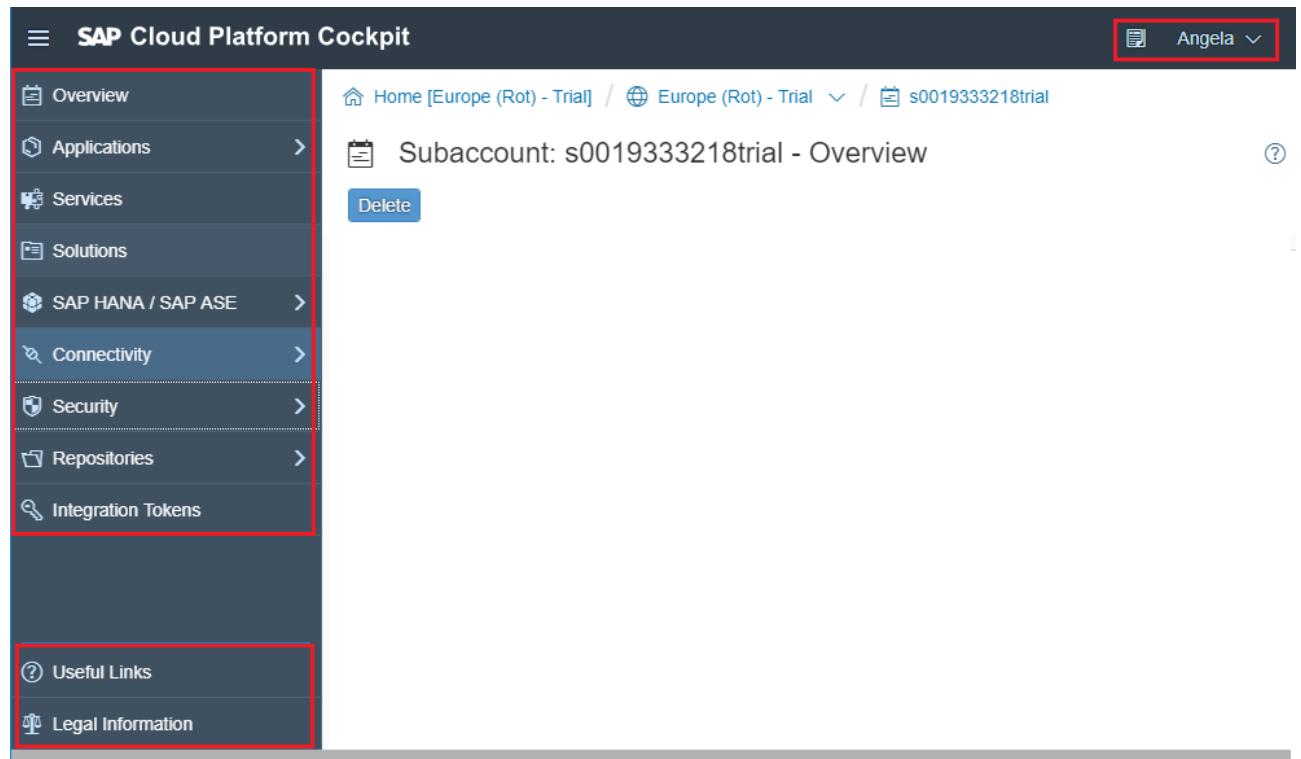


Abbildung 3: SAP Cloud Platform Cockpit der Neo-Umgebung

SAP empfiehlt die Cloud Foundry Umgebung für die Entwicklung von 12-Faktor- und Mikrodienst-basierten Anwendungen, für Szenarien des Internets der Dinge und für maschinelles Lernen. Auch native SAP HANA Anwendungen lassen sich in Cloud Foundry entwickeln [22 S. 11]. Aufgrund der offenen Standardprotokolle ist eine Applikation, die in SAP Cloud Foundry entwickelt wurde, auch bei anderen Anbietern von Cloud Foundry lauffähig. Dies wird durch die Zertifizierung der SAP Cloud Plattform durch die *Cloud Foundry Stiftung* garantiert [27] und verringert den bei SAP bisher hohen Grad der Abhängigkeit vom Anbieter (vendor lock-in). Im Standard werden Programmiersprachen wie Java und NodeJS [22 S. 11] angeboten, Cloud Foundry bietet dem Nutzer zusätzlich die Möglichkeit weitere Programmiersprachen mit Hilfe so genannter *Buildpacks*, welche von

der Stiftung oder der Nutzergemeinschaft (*Community*) angeboten werden, einzubinden [28].

Die zweite, *Neo* genannte, Umgebung ist die Weiterentwicklung der SAP NetWeaver Cloud, eines frühen PaaS von SAP, welches seit 2011 existiert [29] und im März 2013 in die SAP Cloud Platform integriert wurde [30]. Mit ihr kann ein Kunde Java- [31], SAP-HANA-XS- und HTML5-Anwendungen [22 S. 11] in der Cloud entwickeln und betreiben.

Wie Abbildung 4 zeigt, bietet SAP in der SAP Cloud Plattform für jede Umgebung verschiedene Rechenzentrumsstandorte (Regionen) an, welche der Nutzer wählen kann um die Performanz seiner Anwendungen (Antwortzeit, Latenz) zu optimieren [22 S. 7]. In der *Neo*-Umgebung basieren alle Regionen auf Infrastruktur von SAP. In der Cloud-Foundry-Umgebung greift SAP für den weltweiten Betrieb auf Infrastruktur (IaaS) unterschiedlicher Cloud-Anbieter wie *Amazon Web Services*, *Microsoft* und *Google* zurück. Zum Beispiel wird die Region *US East* auf Infrastruktur von Amazon Web Services betrieben. Einige Regionen sind noch im *Beta*-Stadium. Das heißt, dass sie sich noch in einer Testphase befinden und noch nicht für den produktiven Einsatz geeignet sind.

The screenshot displays the SAP Cloud Platform interface with three main sections:

- Cloud Foundry Environment:** Shows four regions:
 - Europe (Frankfurt) - Infrastructure: AWS
 - Europe (Netherlands) - Infrastructure: Azure
 - US Central (IA) Beta - Infrastructure: GCP (BETA)
 - US East (VA) - Infrastructure: AWS
- Neo Environment:** Shows nine regions:
 - Australia (Sydney) - Infrastructure: SAP
 - Brazil (São Paulo) - Infrastructure: SAP
 - Canada (Toronto) - Infrastructure: SAP
 - Europe (Amsterdam) - Infrastructure: SAP
 - Europe (Rot) - Infrastructure: SAP
 - Japan (Tokyo) - Infrastructure: SAP
 - KSA (Riyadh) - Infrastructure: SAP
 - Russia (Moscow) - Infrastructure: SAP
 - UAE (Dubai) - Infrastructure: SAP
 - US East (Ashburn) - Infrastructure: SAP
 - US East (Sterling) - Infrastructure: SAP
 - US West (Chandler) - Infrastructure: SAP
- Trial:** Shows one trial region:
 - Europe (Rot) - Trial - Infrastructure: SAP

Abbildung 4: Regionen in der SAP Cloud Platform

2.3.3 Dienste

Mit der Navigation zum Menüpunkt *Services* erreicht der Nutzer die in der aktuell gewählten Umgebung vorhandenen Dienste. Wie bereits in Abschnitt 2.3 erwähnt, bietet die SAP Cloud Platform diese in zwölf Kategorien an, welche in Abbildung 5 aufgelistet werden. Die grünen Ziffern geben die Anzahl der zum Zeitpunkt dieser Arbeit verfügbaren Angebote in dieser Kategorie an. Eine detaillierte Erläuterung aller Dienste findet sich in der Dokumentation der SAP Cloud Platform [22] oder auf der Internetseite [32].



Abbildung 5: Dienstkategorien auf der SAP Cloud Plattform [32]

Obwohl dieser Bereich für diese Arbeit nur von geringer Bedeutung ist, lohnt es sich, kurz auf den Bereich *Data & Storage* einzugehen. SAP bietet mittlerweile neben zwei HANA-Diensten auch Unterstützung für eigene (*SAP ASE*) und fremde (*PostgreSQL*) relationale Datenbanken für die *NoSQL*-Datenbanken *MongoDB* und *Redis*, einen Dokumenten- und ein Objektspeicherdiensst und Dienste zum Umgang mit großen, komplexen Datenmengen an. Weil der Fokus der Plattform nicht mehr auf HANA liegt, wurde dieser Begriff auch aus dem Namen der Cloud Platform gestrichen.

Für Entwickler besonders interessant ist die Kategorie *Dev Ops*. Diese enthält unter anderem eine integrierte Entwicklungsumgebung (IDE), einen Übersetzungsdiensst und ein Hilfsmittel zur Quelltextverwaltung. Unter Dev Ops versteht man einen Ansatz zur Optimierung des Entwicklungsprozesses [33]. Mit der *SAP Web IDE Full-Stack* (Web IDE), können alle Ebenen einer Anwendung entwickelt werden können, also zum Beispiel ein SAP-HANA-basiertes Modell, Geschäftslogik (in Java) und eine Fiori- bzw. HTML5-Oberfläche. Auf der SAP Cloud Platform findet sich eine zweite Version, die *SAP Web IDE* (ohne Zusatz). Seit dem 14. Mai 2018 rät SAP dazu, alle Projekte aus der einfachen Web IDE in die Full-Stack-Version zu migrieren und diese auch ausschließlich für Neuentwicklungen zu nutzen. Die dritte Version, die *SAP Web IDE for SAP HANA*, steht nicht auf der SAP Cloud Platform zur Verfügung, sondern wird zusammen mit SAP HANA ausgeliefert und kann genutzt werden um native HANA-Apps zu entwickeln [34].

Die SAP Web IDE Full Stack wird alle 14 Tage aktualisiert. Für Offline-Entwicklungen steht die *Web IDE Personal Edition* als lokale Installation zur Verfügung [35]. Diese ist als Ergänzung gedacht und kann nur zur Entwicklung von Oberflächen für Webbrowser und mobile Endgeräte verwendet werden. Sie unterliegt einigen weiteren Einschränkungen [36]. Zudem ist sie nur Nutzern vorbehalten, die auch Zugang zum Web-IDE-Dienst auf der SAP Cloud Platform haben.

Oberflächen von Anwendungen können mit Hilfe von Ressourcendateien in verschiedenen Sprachen angezeigt werden. Zur Erstellung von Übersetzungen bietet SAP den *Translation Hub* an. In diesem werden Übersetzungsprojekte angelegt, welche die Quelldatei und die Zielsprachen definiert und die Übersetzungen mit Hilfe von mehrsprachigen Datenbanken erstellt [37].

Der *Git*-Dienst unterstützt Entwickler bei der Verwaltung von Quelldateien. Versionen können in Git miteinander verglichen, zusammengefügt oder zurückgesetzt werden.

In der Kategorie Integration werden in dieser Arbeit der *OData Provisioning Service* und der Verbindungsdiensst (*Connectivity*) verwendet. Letzterer und der damit in Zusammenhang stehende *SAP Cloud Connector* werden in Abschnitt 2.3.5 behandelt.

Der OData-Provisioning-Dienst ermöglicht das Bereitstellen von OData-Diensten, welche dann in Anwendungen aufgerufen werden können. Vorteil der Verwendung von OData-Diensten für SAP-Webanwendungen ist, dass der Webentwickler keine SAP-Spezifika verstehen muss. Es genügt, wenn er sich mit OData auskennt [38 S. 45]. Das *Open-Data-Protokoll* (OData) dient zur offenen Datenübertragung und basiert auf dem Programmierparadigma *REST*, welches durch folgende sechs Kriterien definiert wird [38 S. 66]:

- Die Architektur muss aus Client und Server bestehen.
- Client und Server müssen durch eine einheitliche Schnittstelle getrennt werden, Ressourcen werden nach einheitlichen Verfahren identifiziert, abgefragt und manipuliert.
- Jede Anfrage des Clients muss alle Daten beinhalten, die der Server zur Verarbeitung braucht, da dieser keine Daten speichert.
- Anfragen sollen gepuffert werden um die Performanz zu erhöhen, die Pufferdauer muss aber spezifiziert werden, um zu vermeiden, dass Clients mit veralteten Daten arbeiten.
- Für den Client spielt es keine Rolle, ob der direkt oder über einen Mittler mit dem Server verbunden ist, der die Daten für die Antwort auf seine Anfrage liefert.
- Optional können Server ausführbaren Code an Clients ausliefern um deren Funktionsumfang zu erweitern. Ein Beispiel hierfür ist die Übermittlung von Java-Script-Befehlen bei der Anzeige einer HTML-Seite.

OData kann auf unterschiedliche Quellen zugreifen und Daten mischen und zusammenführen, ist über alle Versionen vollständig abwärtskompatibel und der Grad der Implementierung ist stets dem tatsächlichen Bedarf angepasst. Eine nachträgliche Erweiterung ist möglich.

Die REST-Kommunikation erfolgt mit in Form folgender Befehle (Operationen):

- **GET:** Ermitteln eines einzelnen Eintrags oder einer Liste von Einträgen
- **POST:** Erzeugen eines neuen Eintrags
- **PUT:** Verändern eines existierenden Eintrags
- **DELETE:** Löschen eines existierenden Eintrags
- **PATCH:** Aktualisieren einzelner Attribute eines existierenden Eintrags.

SAP nutzte OData-Dienste bereits bevor es die SAP Cloud Platform gab. ECC-Systeme sind in der Lage, OData-Dienste zu definieren, welche dann über einen Zugangsserver veröffentlicht und in Webanwendungen angesprochen werden können. Mit Nutzung des OData-Provisioning-Dienstes verlagert sich die Veröffentlichung des OData-Dienstes in die Cloud. In Kapitel 4.1.2 wird das Einrichten eines OData-Dienstes und dessen Veröffentlichung erläutert.

In der Kategorie *User Experience* stehen Hilfsdienste zur Verbesserung des Nutzererlebnisses zur Verfügung. Mit dem *UI Theme Designer* können Anwendungen an das Erscheinungsbild der Firma angepasst werden. Außerdem interessant ist der *Portal*-Dienst. Kunden möchten heutzutage in der Lage sein, rund um die Uhr einen Überblick über ihre Verträge und Profile zu erhalten und einfache Transaktionen selbstständig im Internet abzuwickeln. Dies wird ihnen durch das Angebot von *Self-Service-Portalen* ermöglicht. Diese Verlagerung von Serviceangeboten ins Internet entlasten Personal und sparen Kosten [39]. Der Portal-Dienst erleichtert die Erstellung und Veröffentlichung einer ansprechenden Oberfläche sowie die Zuordnung von Anwendungen zu berechtigten Nutzern. Dabei können von SAP-entwickelte Apps zusammen mit eigenen Anwendungen angezeigt werden. Die genaue Vorgehensweise zur Veröffentlichung einer Fiori-Selbstbedienungsseite beschreibt Abschnitt 4.1.4.

2.3.4 Zugang zu Diensten auf der SAP Cloud Plattform und Bezahlung

Es gibt zwei Ansätze zur Nutzung und Abrechnung der SAP Cloud Platform, ein abonnementbasiertes Modell und ein nutzungsbasiertes Modell [40]. Beide haben ihre Berechtigung abhängig von den Anforderungen und Wünschen des Nutzers.

Bei der abonnementbasierten Variante kann der Kunde im Voraus definieren, welche Services er nutzen möchte und wie diese skaliert sein müssen. Während der Abrechnungsperiode kann er diese Einstellungen nicht verändern. Er zahlt den vereinbarten Preis unabhängig davon, ob er die vereinbarten Dienste tatsächlich genutzt hat oder nicht. Für dieses Modell gibt es Pakete mit einem festgelegten monatlichen Preis und einzelne Services, die der Nutzer zusätzlich zu diesen Paketen buchen kann [41]. Die Bezahlung erfolgt am Beginn der Abrechnungsperiode.

Beim nutzungsbasierten Modell profitiert der Nutzer von der Modularität der SAP Cloud Platform, welche es ihm ermöglicht die einzelnen Dienste unabhängig voneinander zu nutzen. Einzelne Module lassen sich dann buchen, wenn sie benötigt, und abmelden, sobald sie überflüssig werden. Der Kunde hat Zugang zu einem Portal in dem er die Module verwalten und sich über ihre Nutzung und Auslastung informieren kann. Dies ermöglicht ihm eine agile Skalierung seiner Dienste. Die Bezahlung der Services erfolgt nutzungsbasiert und mit *Cloud Credits*, welche einem festgelegten Geldwert entsprechen und zum Vertragsbeginn auf ein Konto eingezahlt werden. Wenn das Guthaben erschöpft ist, muss der Kunde weitere Cloud Credits kaufen um die SAP Cloud Platform weiterhin nutzen zu können.

Beide Bezahlmodelle gelten für eine auf Infrastruktur von SAP betriebene SAP Cloud Platform ab. Es ist ebenso möglich eine SAP Cloud Platform im eigenen Rechenzentrum zu installieren, wenn man die geeignete Infrastruktur bereitstellt. Dies hätte den Vorteil, dass die Kontrolle über die Daten beim Nutzer selbst verbleibt, jedoch hat er dann zusätzlichen finanziellen und zeitlichen Aufwand für die Beschaffung/Verwaltung der Infrastruktur [42] (siehe Abschnitt 11D). Das Bezahlmodell konnte im Rahmen dieser Arbeit leider auch trotz mehrfacher Nachfrage bei verantwortlichen Personen nicht ermittelt werden.

Für Erstnutzer steht eine kostenlose Testversion der SAP Cloud Platform zur Verfügung, welche es ermöglicht einige Services auszuprobieren, und zum Beispiel erste Applikationen in HTML5 und Java zu entwickeln. SAP bietet kostenlose Online-Schulungen an, welche zu großen Teilen mit der Testvariante der Plattform verfolgbar sind. In der Testversion kann der Kunde sich bereits einen Überblick darüber verschaffen, welche Dienste er braucht. Der Zugriff zur SAP Cloud Platform erfolgt über den Link: <http://hanatrial.on-demand.com>. Dort kann sich ein Nutzer mit einem bestehenden SAP-Konto anmelden oder ein neues Konto registrieren. Die Testlizenz ist zeitlich unbefristet. Eine Tabelle zur Übersicht über die Einschränkungen in den Testkonten von Cloud-Foundry- und Neo-Umgebung befindet sich in Abschnitt 11E. Zum Beispiel ist es in der Neo-Umgebung nicht möglich, weitere Personen zum Testkonto hinzuzufügen und es können nur zwei lokal-installierte Systeme mit der SAP-Cloud Platform verbunden werden.

2.3.5 Anbindung an andere Systeme

Die Cloud Platform bietet für die Vernetzung von Systemen einen großen Vorteil: Anstatt je zwei Systeme durch Punkt-zu-Punkt-Verbindungen zu verknüpfen und damit eine unübersichtliche Menge von zu verwaltenden Verbindungen zu schaffen, stellt die Cloud Plattform einen zentralen Verbindungsknoten dar. Jedes Endgerät wird einmal an die Cloud Platform angebunden und steht damit automatisch für jedes andere, mit der Cloud Platform verbundene, Endgerät zur Verfügung. Wie Abbildung 6 zeigt, ist die SAP Cloud Platform in der Lage mobile, lokale und Cloud-Anwendungen von Geschäftspartnern und öffentlichen Behörden sowie soziale Netzwerke zu verknüpfen.

Die SAP Cloud Plattform unterstützt in der Neo-Umgebung folgende Protokolle: *HTTP, SMTP, IMAP, POP3, LDAP, TCP* und *RFC*. In der Cloud-Foundry-Umgebung kann nur HTTP genutzt werden [22 S. 28f]. Die Protokolle IMAP, POP3 und SMTP dienen zum Senden und Empfangen von E-Mails. Über LDAP können zum Beispiel Nutzerverwaltungssysteme wie das Active Directory von Microsoft angeschlossen werden. Die Verbindung zu On-Premise-Systemen von SAP oder anderen Anbietern kann über HTTP, TCP und RFC erfolgen. HTTP ermöglicht zusätzlich die Verbindung zu Internetdiensten (APIs). RFC kann genutzt werden um Funktionsbausteine in SAP-Systemen aufzurufen.

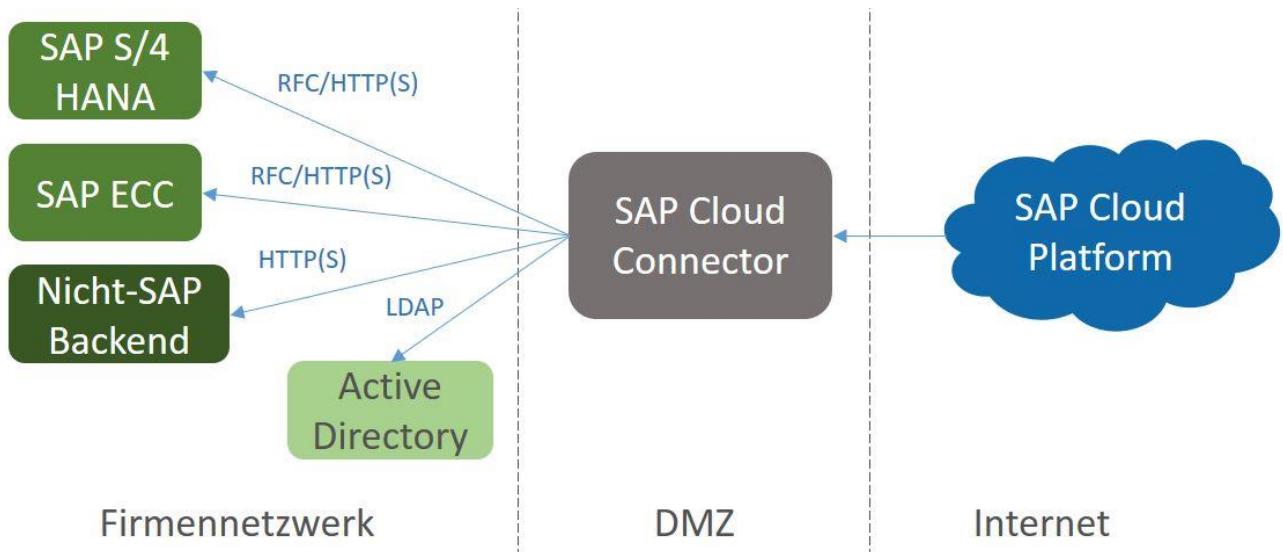


Abbildung 6: Schematische Darstellung der Integrationsmöglichkeiten von SAP Cloud Platform [43]

In der Dokumentation der SAP Cloud Platform findet man Hilfsmittel zur Unterstützung von Integration von und Verbindungen zu Drittsystemen. Einer von Ihnen ist von besonderer Bedeutung für diese Arbeit: der Cloud Connector Dienst.

Dabei handelt es sich um einen Hilfsdienst von SAP, der einen sicheren Tunnel zwischen lokalen Systemen und der SAP Cloud Plattform schafft, ohne die Firewall für potenziell schädliche Angriffe zu öffnen. Um ihn nutzen zu können, ist eine kostenlose Software

von SAP vorgenommen, welche als installierbare und portable Version zur Verfügung steht. Der Sicherheitsaspekt des Konnektors wird in Abschnitt 3.6 näher beleuchtet, mit der Einrichtung beschäftigt sich der Abschnitt 4.1.1.

2.3.6 Verfügbarkeit und Wartung

Bei der Entscheidung über die Verwendung eines Cloud-Dienstes für unternehmerische Zwecke spielt die Frage nach der Verfügbarkeit und der Wartung eine große Rolle. Mit Abgabe der Verantwortung für die Einrichtung und den Betrieb der Infrastruktur verliert der Nutzer auch die Kontrolle darüber, wann die Anwendung gewartet wird, ob er darüber vorher informiert wird und wie schnell die Behebung eines Ausfalls vorgenommen wird. Daher haben sich so genannte Dienststufenvereinbarungen (*Service Level Agreement*, kurz SLA) eingebürgert. Diese legen genau fest, in welchem Rhythmus ein Dienst gewartet wird und wie lange er dafür ausfallen darf. Ebenso verpflichtet sich der Anbieter zu einer prozentual festgelegten Verfügbarkeit des Dienstes und zu einem Zeitfenster, welches ihm für die Behebung von Ausfällen zur Verfügung steht. Die Dienststufen hängen von der Wichtigkeit des Dienstes ab.

Für die SAP Cloud Platform gelten die von SAP definierten SLA für Clouddienste und eine spezielle Ergänzung. Beide Dokumente finden sich im Internet [44]. Von den Vereinbarungen ausgeschlossen werden Hilfsprogramme, welche der Nutzer herunterlädt und im eigenen Umfeld installiert. Dazu gehören zum Beispiel die SAP Web IDE Personal Edition und der SAP Cloud Connector. Ebenfalls ausgenommen werden *Beta-Dienste*, da diese sich noch in einer Testphase befinden.

CLOUD SERVICE	Regelmäßige Wartungsfenster Anfangszeit in UTC nach Region	Dauer	Wichtige Upgrades Zeitraum in UTC nach Region
SAP Cloud Platform	Keine Ausfallzeit für Wartung erforderlich	-	bis zu viermal im Jahr APJ: FR 14:00–18:00 Uhr, UTC Europa: FR 22:00 Uhr bis SA 02:00 Uhr, UTC Amerika: SA 04:00–08:00 Uhr, UTC
SAP Cloud Platform API Management	APJ: SA 15:00 Uhr, UTC Europa: SA 22:00 Uhr, UTC Amerika: SO 04:00 Uhr, UTC	2 Stunden	bis zu viermal im Jahr APJ: FR 14:00 bis SA 14:00 Uhr, UTC Europa: FR 22:00 bis SA 22:00 Uhr, UTC Amerika: SA 04:00 bis SO 04:00 Uhr, UTC
SAP Cloud Platform Gamification	alle zwei (2) Wochen: APJ: SA 15:00 Uhr, UTC Europa: SA 22:00 Uhr, UTC Amerika: SO 04:00 Uhr, UTC	1 Stunde	bis zu viermal im Jahr APJ: FR 14:00–18:00 Uhr, UTC Europa: FR 22:00 Uhr bis SA 02:00 Uhr, UTC Amerika: SA 04:00–08:00 Uhr, UTC

Abbildung 7: Wartungsfenster und Upgrade-Frequenz für Cloud-Dienste

Bei Nichterfüllung der Versprechungen durch den Anbieter, hat der Nutzer ein Anrecht auf eine Gutschrift. Diese muss er jedoch rechtzeitig beantragen und außerdem muss der Ausfallgrund im Verantwortungsbereich von SAP liegen und darf nicht rechtzeitig (mindestens

5 Tage vorher) angekündigt worden sein. Über die prozentuale monatliche Systemverfügbarkeit erfolgt ein monatlicher Bericht durch den Anbieter.

Die generelle Systemverfügbarkeit von Clouddiensten legt SAP auf 99,5% fest. Im ergänzenden Dokument für die Cloud Platform werden sogar 99,9% genannt und einzelne Services, wie der Dokumentendienst ausgeschlossen. Außerdem gibt es eine Hochverfügbarkeitsoption (*High Availability*) für HANA- und ASE-Dienste, bei welcher mehrere Instanzen des Dienstes gemeinsam genutzt werden und dadurch eine höhere Verfügbarkeit (99,95%) ermöglichen.

Von den Verfügbarkeitsversprechen ausgeschlossen sind regelmäßige Wartungsfenster und Zeiten für wichtige Upgrades (Ausschnitt siehe Abbildung 7). Auch die gesamte Testversion der SAP Cloud Platform unterliegt nicht den Dienststufenvereinbarungen.

Den aktuellen Status wichtiger Komponenten jeder Region veröffentlicht SAP im Internet unter <https://sapcp.statuspage.io/>.

3 Vorüberlegungen zur Implementierung einer Webanwendung

Aufbauend auf den vorher erarbeiteten theoretischen Grundlagen sollen nun die Anforderungen für die zu entwickelnde Anwendung und die Herangehensweise an die Evaluierung der SAP Cloud Platform festgelegt und beschrieben werden.

3.1 Bestehende Anwendung mit Eigenentwicklung

Die Wahl der Eigenentwicklung, welche mit Hilfe der SAP Cloud Platform umzusetzen ist, fiel auf das Szenario *Geschäftspartner anzeigen*. Im ECC-System liegen Daten über alle Kunden eines Energieversorgers vor. Diese heißen bei SAP Geschäftspartner und besitzen Stammdaten wie Adressen oder ihre Bankverbindungen. Für Energieversorger gibt es zwei Arten von Geschäftspartnern: Einzel- und Bündelkunden. Einzelkunden sind Endverbraucher, die im Regelfall eine einzige Adresse haben für die sie Energie vom Versorger beziehen. Als Bündelkunden bezeichnet man eine „Einkaufsgemeinschaft, die ihren Strombedarf für mehrere Abnahmestellen von einem einzelnen Lieferanten decken lässt (z. B. bundesweite Energieversorgung von Filialen eines Unternehmens durch einen einzigen Versorger)“ [45]. Im ECC-System ist diese Unterscheidung nach Kundenart nicht standardmäßig vorgesehen und musste daher in einer Eigenentwicklung umgesetzt werden. (Geschäftspartner in SAP ganz viele Unterscheidungen, wichtig zur Anzeige bestimmter Felder und Ausführung bestimmter Funktionen; Anforderung einzelner Energiekunden war Unterscheidung in Bündel- und Einzelkunde.)

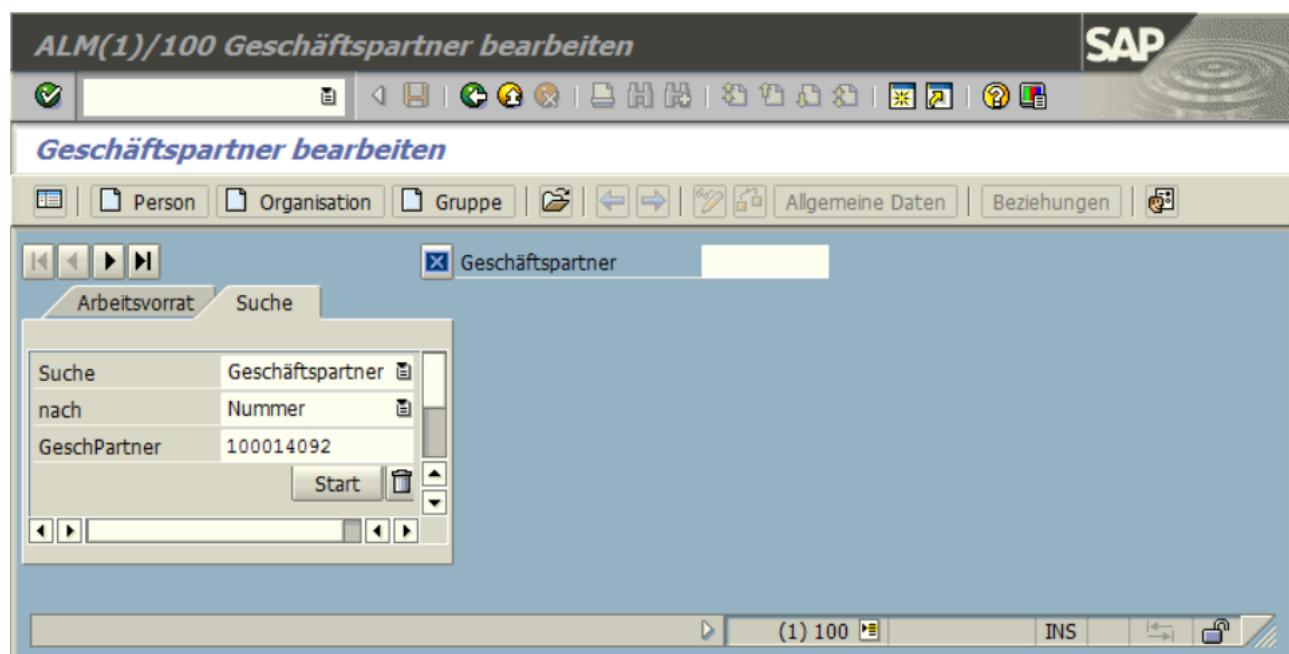


Abbildung 8: Startbild der Transaktion "BP"

Im ECC-System erfolgt die Anzeige des Geschäftspartners unter anderem mit der Transaktion *BP*. Eine Transaktion ist eine Buchstabenfolge, mit der ein Programm schnell aufgerufen werden kann. Abbildung 8 zeigt den Startbildschirm dieses Programms in dem für SAP traditionellen Stil. Links unter der zweiten Menüleiste befindet sich eine graue Box mit zwei Reitern. Der Reiter *Suche* dient dazu eine Auswahl von Geschäftspartnern als Liste auszugeben. Als Suchkriterium kann hier der Name, die Nummer oder die Adresse des Kunden gewählt werden. Führt man einen Doppelklick auf einen der Treffer durch, werden die Stammdaten im rechten Teil des Bildschirms in acht Reitern angezeigt (siehe Abbildung 9). Der vierte Reiter mit dem Namen *Steuerung* beinhaltet die Information über die Kundenart, also darüber ob der Geschäftspartner ein Einzel- oder Bündelkunde ist.

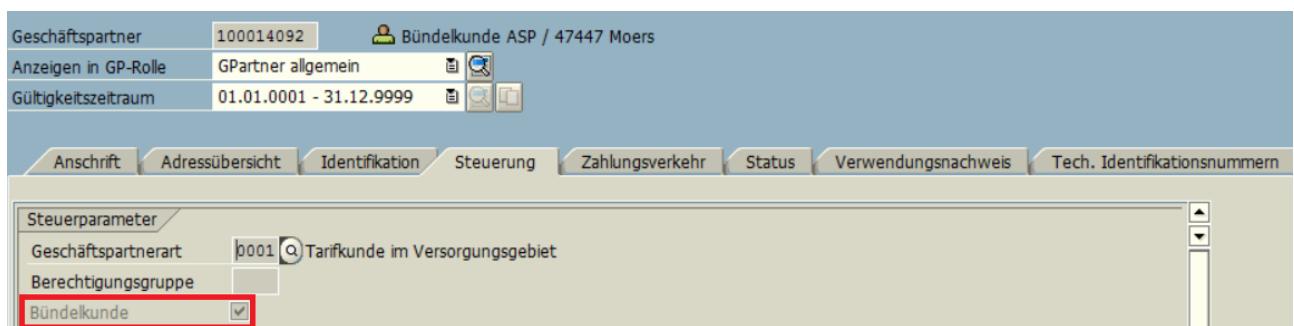


Abbildung 9: Ansicht der Transaktion *BP*

Die zu entwickelnde Webanwendung soll in vereinfachter Weise die Funktion der Transaktion *BP* abbilden. Ein Geschäftspartner soll anhand seiner Nummer auffindbar sein und mit den Stammdaten (Name, Vorname, Kundenart, Vertragskonten) angezeigt werden.

3.2 Anforderungen an die Anwendung

Vor der Umsetzung der Webanwendung müssen einige Anforderungen an die Anwendung, wie ihre Funktionalität, Zielgruppe und die Laufzeitumgebung definiert werden.

Bei der Funktionalität unterscheidet man zwischen Muss-, Soll- und Kann-Kriterien. Erstere müssten zwingend erfüllt sein, damit die App einen Nutzen bringt, während es sich bei Soll-Kriterien um die Beschreibung weiterer geplanter Ausbaustufen handelt. Kann-Kriterien geben einen Ausblick auf weitere Funktionalitäten, die in der Anwendung eingebaut werden können, was jedoch nicht Teil des aktuellen Auftrags ist [46]. Der Kriterienkatalog wird aus Abbildung 10 erkenntlich.

Zwingend notwendig für eine sinnvolle Anwendung ist die Anzeige von Geschäftspartnern als Liste. Diese ist nach der Partnernummer durchsuchbar. Bei Klick auf einen Geschäftspartner erscheint eine Detailansicht mit den grundlegenden Stammdaten Nach-

name, Vorname, Kundenart und Partnernummer. Zudem werden die dem Partner zugeordneten Vertragskonten angezeigt. Es erfolgt eine Navigation zu den einzelnen Vertragskonten.

Im nächsten Schritt kann die Anzeige von Rechnungen und Verträgen, das anhängen und Löschen von Dokumenten und die Bearbeitung von Stammdaten implementiert werden. Dies ist jedoch nicht Teil dieser Arbeit. Gleiches gilt für die Erfassung von Zählerständen zum passenden Vertrag und die Anzeige von Verbrauchsdaten als Bericht, aktuelle Auskunft und als Vorhersage. Zielgruppe der Anwendung zur Anzeige von Geschäftspartnern sind Mitarbeiter des Energieversorgers, welche die Verträge oder Stammdaten von Kunden ansehen, verwalten oder ändern müssen. Mit Vergabe passender Berechtigungen ist dafür zu sorgen, dass Mitarbeiter nur die Aktionen ausführen können, die ihnen im Rahmen ihrer Tätigkeit erlaubt sind.

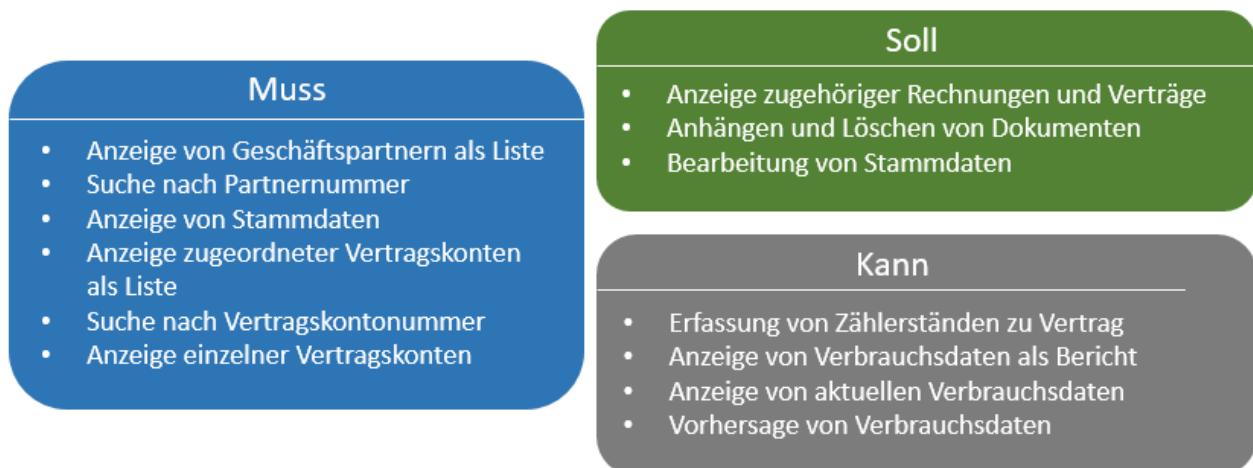


Abbildung 10: Kriterien zur Umsetzung der Webanwendung

Die Anwendung soll eine Fiori-App und damit automatisch in Webbrowsern und auf verschiedenen Endgeräten lauffähig sein. Dennoch gibt es in der Anzeige Unterschiede, welche im Programmcode berücksichtigt werden müssen. In dieser Arbeit soll die Anwendung zur Anzeige im Webbrowser von Desktops und Laptops, insbesondere in Google Chrome optimiert werden. Der Aufruf der Anwendung erfolgt von der Fiori-Startseite, dem Launchpad.

3.3 Fiori

Hinter dem Namen Fiori verbirgt laut Aussage von SAP „kein Produkt, sondern ein Paradigmenwechsel in der Anwendungsentwicklung!“ [47 S. 18]. Es handelt sich um eine „Art, wie Anwendungen bei SAP implementiert werden“ [47 S. 25], um sie besser als die bisherige

Standardoberfläche SAP GUI an die heutigen Bedürfnisse von Softwarenutzern anzupassen. Eine Fiori-Anwendung zeigt dem Nutzer rollenbasiert nur die Daten und Funktionen an, die er benötigt. Die Applikationen sind intuitiv zu bedienen und bilden nur je eine Funktion ab. Alle Fiori-Apps sind in einem einheitlichen Stil gehalten und können auf jedem vom Nutzer gewählten Endgeräts (Laptop, Smartphone, Tablet, Smartwatch) aufgerufen werden.

Fiori-Anwendungen bestehen aus einer Oberfläche (*Frontend*) und einem Datenlieferanten (*Backend*). Für die Oberflächenentwicklung kommt das SAP UI Development Toolkit for HTML5 (SAPUI5) zum Einsatz. Dieses wurde von SAP entwickelt um die Implementation von browserbasierten Anwendungen trotz der Vielfalt an mobilen Plattformen zu ermöglichen [47]. Es basiert auf HTML5 und verwendet CSS3 zur visuellen Gestaltung. SAPUI5 ist *Model-View-Controller*-basiert [47 S. 360]. Darunter versteht man, dass die Anwendung in die drei Module Modell (*model*), Oberfläche (*view*) und Steuerung (*controller*) unterteilt ist. Das Modell enthält Daten und Geschäftslogik, die Oberfläche zeigt Daten an und ermöglicht Nutzerinteraktionen, wie das Betätigen einer Schaltfläche, und die Steuerung dient zur Vermittlung zwischen beiden und reagiert auf Nutzerinteraktionen [48 S. 29]. Neben SAPUI5 gibt es noch OpenUI5, eine quelloffene Lösung, die in den Kernkomponenten exakt mit SAPUI5 übereinstimmt. Da diese jedoch nicht alle Teile der Bibliothek enthält, empfiehlt sich im direkten SAP-Umfeld die Nutzung der proprietären Version [49].

Die Oberfläche einer Fiori-Anwendung unterliegt der SAP Fiori Design Guideline [50]. Ihre Anzeige erfolgt üblicherweise im Fiori-Launchpad. Damit wird eine Startseite im Browser bezeichnet, welche alle Anwendungen anzeigt, für die ein Nutzer berechtigt ist. Die Verbindung zwischen dem Frontend und dem zum Datenlieferanten schaffen OData-Dienste, welche im Zugangsserver veröffentlicht werden, nachdem sie beim Datenlieferanten modelliert wurden [51]. Jede Anwendung, die diesen Kriterien entspricht, darf *Fiori-ähnlich* genannt werden [7].

SAP unterscheidet bei den durch das Unternehmen selbst zur Verfügung gestellten Fiori-Anwendungen zwischen *transaktionalen Anwendungen*, *analytischen Anwendungen* und *Infoblättern*. Unter transaktional versteht man, dass die Anwendung eine Aufgabe abbildet. Diese Fiori-Applikationen können mit jeder darunterliegenden Datenbank genutzt werden. Analytische Applikationen „sammeln Kennzahlen und zeigen sie im Browser an“ [7]. Infoblätter enthalten Stammdaten von zentralen Objekten und zeigen diese kontextbezogen an. Sie ermöglichen eine Navigation zu anderen, im Zusammenhang stehenden Infoblättern. Sowohl analytische Anwendungen als auch Infoblätter sind an die Nutzung einer SAP-HANA-Datenbank gebunden.

Wie bereits in Kapitel 1.2 erwähnt, können Fiori-Benutzeroberflächen auch für ECC-Systeme implementiert werden, dies trifft jedoch speziell nur auf transaktionale Anwendungen zu. Die inhaltlichen Anforderungen beschreiben, dass Stammdaten zu Kunden von Energieversorgern angezeigt werden sollen. Dementsprechend wäre die Anwendung als Infoblatt einzuordnen und nur mit Hilfe einer SAP-HANA-Datenbank betreibbar. Um eine transaktionale App zu erhalten, muss eine spezielle Aufgabe in Bezug zum Objekt *Geschäftspartner* gewählt werden, zum Beispiel die Adressänderung oder die Änderung der Kundenart. Im nächsten Abschnitt wird geprüft, ob eine transaktionale Anwendung existiert, welche im Zusammenhang mit einer speziellen Aufgabe die Anzeige von ausgewählten Stammdaten anzeigt.

3.4 Verfügbarkeit und Erweiterung von Standardfunktionen

Verfügbare Fiori-Anwendungen können in der *Fiori Apps Reference Library* unter <https://fioriappslibrary.hana.ondemand.com> gesucht werden. Die Filterbedingungen sind für diesen Fall wie folgt festzulegen:

- **Product Suite:**
 - SAP Business Suite (ECC)
- **Required Back-End Product:**
 - SAP ERP
 - SAP ERP, min. EHP 7
 - SAP ERP, min. EHP 8
- **Database:**
 - Any DB

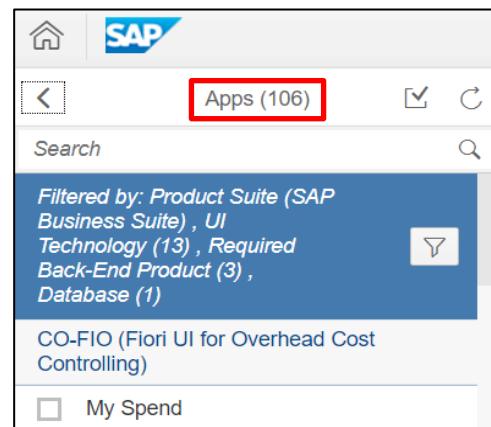


Abbildung 11: Bildschirmaufnahme der Filterung von SAP-Fiori-Anwendungen

Damit werden Applikationen angezeigt, die auf ECC-Systemen (*Business Suite*) ohne SAP-HANA-Datenbank lauffähig sind. Als Ergebnis werden nun 106 Anwendungen angezeigt welche mit Hilfe von Suchbegriffen weiter eingeschränkt werden können. Eine Suche nach *business*, *partner*, *bp*, *customer* oder *address* liefert kein passendes Ergebnis. Entfernt man alle Filter und sucht nach *business partner*, erhält man 137 Anwendungen als Ergebnis, darunter die Applikation *Display Business Partner (BUP3)* [52], welche jedoch nur für S/4-HANA-Systeme mit HANA-Datenbank geeignet ist.

Weil keine passende Standardanwendung gefunden werden konnte, ist im Rahmen dieser Arbeit eine eigene energiewirtschaftliche Anwendung zu entwickeln. Die Einbindung und Anpassung von durch SAP erstellten Fiori-Anwendungen wird nicht gezeigt.

3.5 Architektur der eigenen Anwendung

Für die Eigenentwicklung der Anwendung stehen verschiedene Technologien und mögliche Architekturen zur Verfügung. Die App soll Fiori-ähnlich sein. Das heißt ihre Oberfläche soll auf dem SAPUI5-Framework aufbauen und sich an die Fiori Design Guideline halten. Die Oberfläche soll ihre Daten mit Hilfe eines OData-Dienstes aus dem Backend-System beziehen. Zudem soll die Anwendung in der SAP Cloud Platform entwickelt werden und anschließend auf einem Fiori-Launchpad angezeigt werden.

Wie bereits in Abschnitt 2.3.5 dargelegt, benötigt man einen SAP Cloud Connector um eine Verbindung zwischen der SAP Cloud Platform und lokalen Systemen, hier dem Zugangsserver (*Gateway Hub*), herzustellen. Dieser übernimmt dann die Bereitstellung des OData-Dienstes. Abbildung 12 zeigt diese und eine weitere Verbindungsvariante. Sollte kein separater Zugangsserver zur Verfügung stehen, ist es auch möglich zur Bereitstellung des OData-Dienstes den OData-Provisioning-Dienst der SAP Cloud Platform zu nutzen. Dafür wird im Backend die Software-Komponente *Business Enablement Provisioning* (IW-BEP) [53] benötigt. Beide Varianten werden in dieser Arbeit umgesetzt.

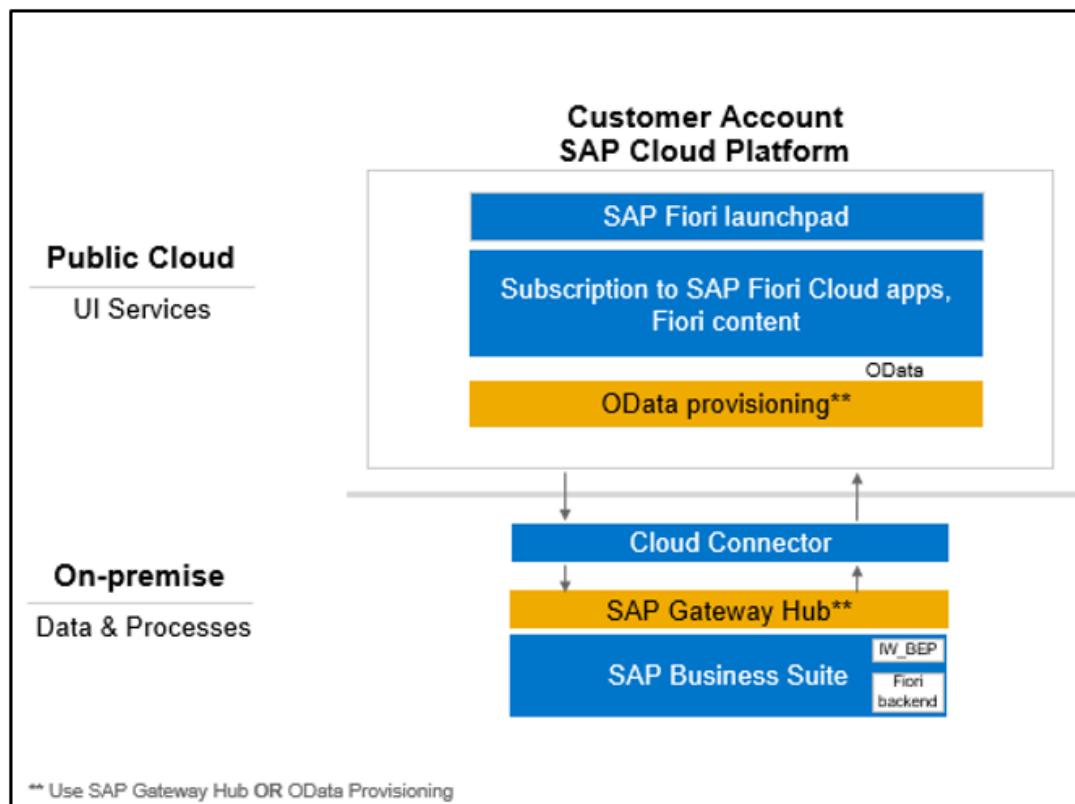


Abbildung 12: Übersicht über die mögliche Architektur von Fiori-Applikationen

Ebenso wäre es möglich, an Stelle des separaten Zugangsservers einen direkt auf dem Backend-Server installierten Zugangsserver zu verwenden (*embedded deployment*) [54]. Dies wird jedoch von SAP als kritisch angesehen, da dadurch bei einer Aktualisierung

der Gateway- oder SAPUI5-Version ein höherer Aufwand entsteht [47 S. 35-36]. Zudem könnte sich ein Angreifer, welcher Zugang zum SAP Gateway Hub erlangt hat, auf alle Daten des Backend-Systems zugreifen.

Als separater Zugangsserver wird der Frontend-Server SGW verwendet, welcher eine SAP-Gateway-Hub-Komponente beinhaltet und per RFC-Verbindung mit dem SAP-ECC-System *SBE* verbunden ist. Dieses wird als Backend verwendet und ist eine Kopie eines bei einem Energieversorger eingesetzten Produktivsystems. Daher sind im System auch Kopien von Kundendaten. Im Rahmen dieser Arbeit soll aus Gründen des Datenschutzes (siehe Abschnitt 3.6) mit anonymen oder fiktiven Kundendaten gearbeitet werden.

Über den Zugangsserver im SGW wurde bereits ein OData-Dienst exponiert, welcher alle Daten enthält, welche für diese Anwendung benötigt werden: Kundennamen, Vertragskonten und die Kundenart. Der Service heißt */ASPSP/ERP_UTILITIES_UMC*.

OData-Dienste für Versorgungsunternehmen werden nicht standardmäßig in SAP-ECC-Systemen ausgeliefert, sondern können Teil von erweiternden SAP-Produkten sein. *SAP Multichannel Foundation for Utilities and Public Sector (MCF)* gibt Versorgungsunternehmen und öffentlichen Behörden die Möglichkeit Self-Service-Portale für ihre Kunden zur Verfügung zu stellen. Kunden eines Energieversorgers können in einer solchen Webanwendung unter anderem Ihre Stammdaten einsehen und ändern oder ihren selbstabgelesenen Zählerstand eintragen. Seit es die so genannten intelligenten Stromzähler (*Smart Meter*) gibt, kann dem Kunden in einem Selbstbedienungsportal auch der aktuelle Verbrauch und eine entsprechende Verbrauchsvorhersage angezeigt werden. Dies stellt einen erheblichen Mehrwert dar. Bei dem Selbstbedienungsportal handelt es sich in der Regel um eine Webanwendung, welche ihre Daten mit Hilfe von OData-Diensten aus dem Backend bezieht. SAP MCF bringt daher eine eigenen OData-Bibliothek mit, welche für energiewirtschaftliche Anwendungen genutzt werden kann. Der erwähnte Dienst */ASPSP/ERP_UTILITIES_UMC* ist eine Anpassung des SAP-Standard-Dienstes *ERP_UTILITIES_UMC* aus der OData-Bibliothek in SAP MCF. Es ist zu prüfen, ob dieser Dienst für die Implementierung in dieser Arbeit verwendet werden kann oder ein neuer OData-Dienst exponiert werden muss.

Nach Abschluss der Entwicklung stehen erneut zwei Varianten zur Verteilung und zur Anzeige der Anwendung zur Verfügung. Bei der Cloud-Variante wird die Anwendung in die SAP Cloud Platform verteilt. Dort kann im Portal-Dienst ein Fiori-Launchpad erzeugt werden, welchem die Anwendung nach erfolgreicher Verteilung in der Cloud zugeordnet werden kann. Diese Variante wird in Abbildung 12 dargestellt. Wählt man die lokale Alternative, wird ein Fiori-Launchpad auf dem Applikationsserver eingerichtet, welchem die dort-hin verteilte Anwendung zugeordnet werden kann. Der Frontend-Server SGW übernimmt in

dieser Arbeit eine Doppelrolle als Zugangs- und Applikationsserver und stellt bereits ein Fiori-Launchpad zur Verfügung, auf welchem die Anwendung angezeigt werden kann. Eine Anleitung zur Einrichtung des Fiori-Launchpads findet sich im Internet [55] oder bei *Englbrecht* [47 S. 253-259]. Auch in diesem Fall werden beide Varianten im Rahmen dieser Arbeit umgesetzt.

3.6 Sicherheit und Datenschutz

Personenbezogene Daten unterliegen besonderem Schutz durch den Gesetzgeber und müssen daher entsprechend vor Missbrauch durch Dritte gesichert werden. Für die in dieser Arbeit entwickelte Anwendung ist es daher wichtig, dass personenbezogene Daten verschlüsselt gespeichert und nur befugten Personen angezeigt werden und bei der Übertragung vom Datenlieferanten zur Oberfläche so gut wie möglich geschützt sind. Dies trifft für die Entwicklungszeit als auch für die produktive Anwendung zu.

Zusätzlich zum deutschen Bundesdatenschutzgesetz existiert seit dem 25. Mai 2018 auch die europäische Datenschutzgrundverordnung (EU-DSGVO). Diese gilt für Unternehmen mit Niederlassung in der *Europäischen Union* (EU), welche personenbezogene Daten verarbeiten, unabhängig davon, ob die Verarbeitung in der EU stattfindet. Zudem gilt es aber auch für Unternehmen mit Sitz außerhalb der EU, wenn diese einer Person in der EU Waren oder Dienstleistungen anbieten oder deren Verhalten beobachten [27]. Arvato Systems Perdata verarbeitet die personenbezogenen Daten, welche ihre Kunden über ihre Kunden speichern müssen um die Energieversorgung zu ermöglichen.

Als deutsches Unternehmen unterliegt ASP in vollem Maße der Datenschutzgrundverordnung und muss sicherstellen, dass Anwendungen, welche dem Kunden zur Verfügung gestellt werden, den Datenschutzrichtlinien entsprechen. Für die Authentifizierung eines Nutzers an SAP-Backend-Systemen stehen verschiedene Authentifizierungsoptionen zur Verfügung, welche teilweise auch *Single Sign-On* (SSO) ermöglichen. SSO bedeutet, dass sich ein Nutzer während einer gültigen Sitzung nur einmal anmelden muss und alle gekoppelten Systeme und Anwendungen ohne weitere Anmeldung nutzen kann [56].

- **Benutzer-ID und Kennwort** (*Basic Authentication*): Dies ist das von ASP für die Backend-Systeme genutzte Szenario. Ein Nutzer wird im System eingerichtet und bekommt ein Initialkennwort zugewiesen, welches er beim ersten Login ändern muss. Eine Umsetzung als SSO ist hiermit nicht möglich. Der OData-Service kann ebenfalls nur mit den Zugangsdaten des entsprechenden Systems erreicht werden.

- **Anmeldetickets:** Ein in der Systemlandschaft befindliches Portal stellt einem Nutzer, der sich dort mit Passwort und Benutzername anmeldet ein Anmeldeticket aus, welches als *Cookie* im Web-Browser des Benutzers abgelegt wird und ihn automatisch bei allen Systemen anmeldet, die Anmeldezertifikate unterstützen.
- **Header-Variablen:** Ein Nutzer meldet sich an einem *Web-Zugriffsverwaltungsprodukt* (WZV) eines Drittanbieters an und erhält eine authentifizierte Benutzerkenntnis als Teil des *HTTP-Headers* zurück. Damit kann er sich anschließend bei Anwendungen eines SAP Anwendungsservers anmelden.
- **Open Authorization (OAuth) 2.0:**

Es handelt sich hier um einen Standard zur sicheren API-Authentifizierung. So kann ein Nutzer (in der Rolle als *Resource Owner*) einer Anwendung (dem *Client*) erlauben, in seinem Namen auf geschützte Ressourcen / Daten auf einem Server (*Resource Server*) zuzugreifen, ohne seine Anmeldedaten dafür bekannt geben zu müssen. Dies nutzen Entwickler einer Anwendung um eine Authentifizierung zu implementieren ohne Anmeldedaten von Nutzern zu speichern.



Abbildung 13: Anzeige der vom Client geforderten Berechtigungen bei OAuth 2.0 [77]

Stattdessen delegieren Sie die Authentifizierung der Nutzer an einen OAuth-Anbieter wie *Google*, *Facebook* oder *Twitter*. OAuth beinhaltet eine detaillierte Einschränkung der Zugriffsrechte. Dem Nutzer wird nach Eingabe der Anmeldedaten angezeigt, welche Berechtigungen der Client fordert (siehe Abbildung 13). Stimmt der Nutzer zu, wird er zurück zur Client-Anwendung geleitet. Hierbei handelt es sich nicht um eine SSO-Lösung. OAuth kann aber mit anderen Diensten gekoppelt werden um SSO zu ermöglichen.

- **Kerberos-Authentifizierung:** Eine zentrale Schlüsselvergabe (*Key Distribution Center*, KDC) dient im Netzwerk als vertrauenswürdige dritte Partei bei der Vermittlung zwischen zwei Systemen. Sie besteht aus einem Authentifizierungsserver und einem Ticketvergabeserver. Ein Nutzer meldet sich an der SAP Cloud Platform an und möchte ein Backend-System erreichen (siehe Abbildung 14). Dafür sendet die SAP

Cloud Platform eine verschlüsselte Nachricht mit der Identität des Users an den SAP Cloud Connector, welcher die Kerberos-Bibliothek enthält.

Dort wird die Nachricht entschlüsselt und der Nutzer bei der KDC abgefragt. Dort findet eine Kommunikation zwischen dem Authentifizierungsserver und dem Ticketvergabeserver statt. Ist diese Abfrage erfolgreich, erhält der Konnektor ein Ticket, welches die Identität des Nutzers bestätigt.

Dieses sendet er in einer Anfrage an das Backendsystem. Dort wird darauf vertraut, dass es sich tatsächlich um den Nutzer handelt, der im Ticket aufgeführt wird und eine Anmeldung erfolgt, falls dieser die entsprechenden Berechtigungen hat.

Einige der der Anmeldeoptionen von SAP-Systemen können auch im Zusammenhang mit der SAP Cloud Platform verwendet werden um eine sichere und nutzerfreundliche Authentifizierung mit SSO zu erreichen.

- **X.509 Client-Zertifikate:** Hat ein Nutzer in seinem Browser ein gültiges X.509-Zertifikat wird er für den Zugriff zu SAP-Systemen authentifiziert, ohne seine Benutzerkennung eingeben zu müssen, wenn das Zertifikat von einer vertrauenswürdigen Stelle ausgestellt wurde und die Systeme für die Verwendung von *Secure Sockets Layer* (SSL) konfiguriert sind.
- **SAML-Browser-Artefakte / SAML 2.0:** SAML ist die Abkürzung für *Security Assertion Markup Language* und bezeichnet einen offenen Standard für die Umsetzung von SSO. Dafür werden ein Dienstanbieter (*Service Provider*) und ein Identitätsanbieter (*Identity Provider*) benötigt.

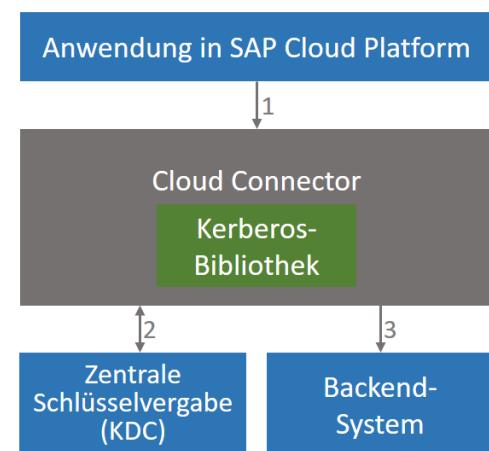


Abbildung 14: Ablauf einer Kerberos-Authentifizierung [23 S. 305]



Abbildung 15: Ablauf einer SAML-Authentifizierung [57]

Möchte sich ein Nutzer am Dienstanbieter anmelden, leitet dieser die Authentifizierungsanfrage an einen Identitätsanbieter weiter, welcher Identitätsinformationen von Nutzern, Rollen und Gruppen (zusammengefasst *Principals*) speichert und prüft, ob diese für den Dienst, von dem die Authentifizierungsanfrage kommt, berechtigt ist (siehe Abbildung 15). Falls der Nutzer sich in der aktuellen Sitzung bereits beim Identitätsanbieter angemeldet hat, entfällt eine weitere Abfrage der Anmeldedaten und der Dienstanbieter bekommt sofort den Nutzer und die Authentifizierungsantwort.

SAP stellt auf der SAP Cloud Platform selbst einen Identitätsanbieter zur Verfügung. Dieser ermöglicht es dem Nutzer auf alle Dienste der SAP Cloud Platform mit einmaliger Anmeldung zuzugreifen. Diese Art der Authentifizierung kann ebenfalls genutzt werden, um einen bereits bei der Cloud Platform authentifizierten Nutzer am Backend-System anzumelden ohne nach den entsprechenden Anmeldedaten zu fragen.

Die von ASP zur Anmeldung mit *Basic Authentication* gehört zu den Anmeldemethoden, die den geringsten Aufwand bei der Einrichtung erfordern. In Zusammenhang mit einer HTTPS-Verbindung können Benutzername und Passwort zwar nicht mehr so leicht bei der Übertragung abgefangen werden, jedoch wird das Passwort im Browser entweder kurzzeitig (um zu verhindern, dass der Nutzer für jede HTTP-Anfrage sein Passwort eingeben muss) oder langfristig (wenn der Nutzer diese Option im Browser wählt) gespeichert. Perspektivisch wäre es daher sinnvoll eine andere Authentifizierungsmethode zu wählen oder die Basic Authentication zu einer *2-Faktor-Authentifizierung* zu verstärken. Möglicherweise kann es sogar für den Nutzerkomfort sinnvoll sein, über die Einrichtung von SSO nachzudenken. Dies ist jedoch nicht Teil dieser Arbeit.

Werden Dritte, wie SAP, in die Dienstleistung involviert, haftet Arvato Systems Perdata auch dafür, dass auch diese die Datenschutzrichtlinien einhalten [28 S. 41]. Da es sich bei SAP ebenfalls um ein deutsches Unternehmen handelt, kann davon ausgegangen werden, dass SAP selbst ein großes Interesse daran hat, für eine Einhaltung des Datenschutzes zu sorgen. Die SAP Cloud Platform ist nur mit einem Konto und passendem Passwort zugänglich. Im Rahmen der SAP Cloud Platform involviert SAP jedoch wiederum Dritte und es ist zu prüfen, ob diese ebenfalls datenschutzkonform arbeiten, da ASP für alle in der Wertschöpfungskette folgenden Dienstleister haftet, falls es zu einem Datenschutzvorfall kommt.

Daher ist bei der Wahl der Region (siehe Abschnitt 2.3.2), welche man in der SAP Cloud Platform benutzen möchte, Vorsicht geboten. Alle Daten der Testzugänge werden in der Region *Europe (Rot)–Trial* verwaltet, liegen also in einem SAP eigenen Rechenzentrum

in Sankt Leon-Rot. Bei einem bezahlten Zugang zur SAP Cloud Platform ist dies abhängig von der gewählten Region. Es stellt sich die Frage, ob amerikanische Großkonzerne wie *Google*, *Amazon* und *Microsoft* ihre Dienstleistungen tatsächlich an die Anforderungen der europäischen Datenschutzverordnung angepasst haben. Klar ist, dass Sie dieser unterliegen, sobald sich die Personen, deren Daten verwaltet werden, in der EU aufhalten [58]. Um so sicher wie möglich zu sein, sind grundsätzlich die SAP-eigenen, europäischen Regionen in der SAP Cloud Platform vorzuziehen. Selbst dann ist es jedoch möglich, dass SAP die Daten zur Verarbeitung in Länder außerhalb des Europäischen Wirtschaftsraums übermittelt. Dies wird in der Datenschutzerklärung festgehalten. In einem solchen Fall werden jedoch Verträge mit den zusätzlichen Verarbeitern abgeschlossen, welche dafür sorgen, dass dennoch das europäische Datenschutzniveau eingehalten wird [59].

Auch in Deutschland können die personenbezogenen Daten durch SAP im Rahmen eines Auskunftsersuchens weitergegeben werden. Dafür muss jedoch entweder die Einwilligung der betroffenen Person vorliegen oder eine ausreichende rechtliche Grundlage. In den Nutzungsbedingungen der SAP Cloud Platform verankert SAP die Einwilligung zur Weitergabe personenbezogener Daten, falls dies rechtlich nötig ist [60].

Einen Angriffspunkt stellt auch der SAP Cloud Connector dar, da er einen Tunnel zwischen dem Internet und dem Zugangsserver, bzw. dem Backend, öffnet. SAP empfiehlt daher den Zugriff auf den Cloud Connector auf die unbedingt notwendigen Personen zu beschränken und ein eigenes System nur für den SAP Cloud Connector zu benutzen [53 S. 266]. Zudem prüft die Software den Sicherheitsstatus des Konnektors in Echtzeit und gibt Ratschläge zur Verbesserung. Die SAP Cloud Platform unterstützt verschiedene Anmeldeverfahren und Lösungen für SSO.

Da die Datenschutzbestimmungen und ihre möglichen Umsetzungen in Webanwendungen äußerst komplex sind, sollen sich die Sicherheitsmaßnahmen dieser Anwendung darauf beschränken, den Zugang zur Anwendung und die Berechtigungen zu beschränken und die Übertragungswege der personenbezogenen Daten zu sichern.

3.7 Entwicklungsumgebung

Zur Entwicklung von Fiori-Anwendungen gibt es mehrere mögliche Entwicklungsumgebungen. Die einfachste Basis dafür ist das SAPUI5-Framework, welches von SAP online [35] zur Verfügung gestellt wird. In der SAP Cloud Platform werden Fiori-Anwendungen in der Neo-Umgebung entwickelt. SAP empfiehlt die Nutzung der dort vorhandenen SAP Web IDE Full-Stack (siehe Abschnitt 2.3.3). Sie bietet folgende Funktionalitäten:

- **Quelltext Bearbeitung:** Mit der SAP Web IDE Full-Stack können freie Quelltexte verfasst werden. Vervollständigung, Validierung und Vorlagen (*Templates*) werden für XML, properties-Dateien, JavaScript und SAPUI5 angeboten. Die Texte sind vollständig durchsuchbar und können automatisch formatiert werden (JavaScript, JSON, XML und CSS). Probleme mit dem Quelltext können in einer eigenen Ansicht angezeigt werden.
- **Testen und Fehlersuche:** Anwendungen können im Browser in verschiedenen Auflösungen und Sprachen ausgeführt werden. Dafür stehen in der Web IDE auch Testdaten zur Verfügung. Um die Anwendung auf einem mobilen Endgerät zu öffnen, kann in der Web IDE ein Quick Response (QR) -Code erzeugt werden, welcher dann mit dem Endgerät gescannt werden muss, um die manuelle Eingabe des Anwendungs-Links zu vermeiden.
- **Unterstützung mehrerer Geräte:** nicht nur die darin entwickelten SAPUI5-Anwendungen, sondern auch die Web IDE selbst kann auf stationären und tragbaren Rechnern, Tablets und Mobiltelefonen ausgeführt werden.
- **Anwendungsvorlagen:** SAPUI5-Anwendungen können mit Hilfe von Vorlagen und Assistenten unter Berücksichtigung der Fiori-Richtlinien erstellt werden.
- **Graphische Oberflächen-Editoren:** SAP Web IDE bietet einen graphischen Layout-Editor und einen UI Adaption Editor. Ersterer dient dazu ein Layout zu entwerfen und Daten an bestimmte Objekte zu binden. Der Quelltext wird dabei automatisch im Hintergrund generiert. *UI Adaption* ermöglicht die Anpassung von unterstützten Fiori-Oberflächen während der Laufzeit.
- **Erweiterungen von Fiori-Anwendungen:** von SAP zur Verfügung gestellte Fiori-Anwendungen können mit Hilfe eines visuellen Erweiterungseditors erweitert werden.
- **Auslieferung von Anwendungen:** Entwicklungen können aus der Web IDE in die Cloud Platform ausgeliefert und in einem Fiori-Launchpad auf der Plattform registriert werden. Ebenso können Anwendungen auf lokale SAP-Server verteilt werden.

Bevor die SAP Web IDE zur Verfügung stand, baute SAP auf die Entwicklung in einer Eclipse IDE. Dort lässt sich das SAPUI5-Framework mit Hilfe einer Erweiterung (*Plugin*) von SAP integrieren. Das Eclipse-Plugin wird jedoch nur noch aktualisiert um mit neuen SAPUI5-Versionen lauffähig zu bleiben. Neue Funktionen der Bibliotheken werden nicht in Eclipse verfügbar sein [47 S. 349]. Theoretisch kann das SAPUI5-Framework in der quelloffenen OpenUI5-Version in jede integrierte Entwicklungsumgebung importiert werden. Die Entwicklung und Anpassung ist daher nicht von der SAP Cloud Platform abhängig. Holger Schäfer [61] nennt neben Eclipse und der SAP Web IDE auch das *SAP HANA Studio in der*

Developer Edition und die *WebStorm IDE* der Firma *JetBrains*. Ersteres ist vor allem bei Entwicklungen für Anwendungen auf SAP-HANA-Basis sinnvoll. Letztere wird von SAP nicht durch ein Plugin unterstützt und die initiale Konfiguration ist daher aufwändiger als bei Eclipse oder SAP Web IDE. Es scheint daher sinnvoll, der Empfehlung von SAP zu folgen und im Rahmen dieser Arbeit die SAP Web IDE Full-Stack zu verwenden. Ein Umstieg auf WebStorm oder eine andere IDE ist zu einem späteren Zeitpunkt möglich.

Um ohne trotz einer IDE in der Cloud auch ohne Internet arbeiten zu können, ist es nötig, lokal die Web IDE Personal Edition zu installieren und beide Entwicklungsumgebungen mit Hilfe von *Git*, einer Software zur Versionsverwaltung, zu verbinden. Eine ausführliche Anleitung hierzu findet sich im Internet [62]. Der Entwickler speichert einen Stand im Git-Verzeichnis, wechselt zur lokalen Entwicklungsumgebung, ruft den Stand darin ab und kann dort weiterarbeiten, wo er Online aufgehört hat. Im Fall eines Ausfalls der SAP Cloud Platform Web IDE steht Offline nur der zuletzt in Git gespeicherte Stand zur Verfügung. Es ist daher empfehlenswert den aktuellen Stand so oft wie möglich in Git zu registrieren.

Für die Verwendung der SAP WebIDE Cloud Version muss nur der Service in der SAP Cloud Platform geöffnet werden. Die Internetadresse kann als Lesezeichen gespeichert und vor dort wieder aufgerufen werden. So erspart sich der Nutzer den Weg über das SAP Cloud Platform Cockpit. Die Personal Edition der Web IDE muss im Internet [35] heruntergeladen und das Verzeichnis in den gewünschten Ordner entpackt werden. Im Unterordner *eclipse* findet sich eine Datei namens *orion.exe*. Öffnet man diese, wird die Laufzeitumgebung gestartet. Anschließend ist die Web IDE Personal Edition über den Link <http://localhost:8080/webide/index.html> erreichbar. Beide Entwicklungsumgebungen sind sofort einsatzbereit und benötigen keine Installation von Erweiterungen.

3.8 Evaluierungskriterien

Im Rahmen der Umsetzung, soll die SAP Cloud Platform evaluiert werden. Dieser Abschnitt stellt die Fragen vor, mit deren Hilfe die SAP Cloud Platform letztendlich untersucht wird. Die Beantwortung der Fragen erfolgt in Abschnitt 4.2. Hauptaugenmerk soll dabei die Machbarkeit bestimmter Aspekte sein. Der zugehörige Fragenkatalog sieht wie folgt aus:

- Ist es möglich ein SAP ECC-System mit der Cloud Plattform zu verbinden?
- Kann darin eine Fiori-App mit Daten aus einem ECC-System entwickelt werden?
- Kann die entwickelte Anwendung ohne Änderungen des Quellcodes für Kunden genutzt werden, deren Geschäftspartner ohne Eigenentwicklungen abgebildet werden können?

Darüber hinausgehend ergeben sich weitere Fragen zur Evaluation der SAP Cloud Platform:

- Welchen Mehrwert stellt die SAP Cloud Plattform für die Entwicklung von Fiori-Apps dar?
- Ist es denkbar, die Cloud Platform auch zum Ausliefern von Anwendungen für den Produktivbetrieb zu nutzen?

4 Evaluierung der SAP Cloud Platform

In diesem Kapitel soll die Anwendung, welche im vorherigen Abschnitt definiert wurde, umgesetzt und die einzelnen Schritte dokumentiert werden. Danach folgt der Vergleich von Diensten der SAP Cloud Platform mit bestehenden Produkten von SAP. Anschließend gibt das Kapitel einen Überblick über das Bewerberumfeld der Cloud Platform und schätzt die Plattform ein.

4.1 Umsetzung der Webanwendung

Bei der Umsetzung (siehe Abbildung 16) werden zuerst das Backend und der Zugangsserver im SAP Cloud Connector angelegt, anschließend der OData-Service im Backend ausgeprägt (1) und über den Zugangsserver beziehungsweise den OData-Provisioning-Dienst exponiert (2). Die SAP Cloud Platform speichert Systeme als so genannte Destinationen. Anschließend wird die Oberfläche in der Web IDE aus einer Vorlage erstellt, der über eine Destination erreichbare OData-Dienst eingebunden und die Anwendung angepasst (3). Die fertige Anwendung wird auf der SAP Cloud Platform und auf den lokalen Applikationsserver verteilt (4) und auf einem Fiori-Launchpad in der Cloud und lokal eingebunden und angezeigt (7). Bei der Dokumentation werden nicht alle, sondern nur besonders wichtige, bzw. Cloud-Platform-spezifische Schritte erfasst. Ausführliche Anleitungen zu vielen Szenarien finden sich unter: <https://blogs.sap.com/>.

Für die weiteren Schritte wird ein Konto auf der SAP Cloud Platform und ein Unterkonto in der Neo-Umgebung benötigt.

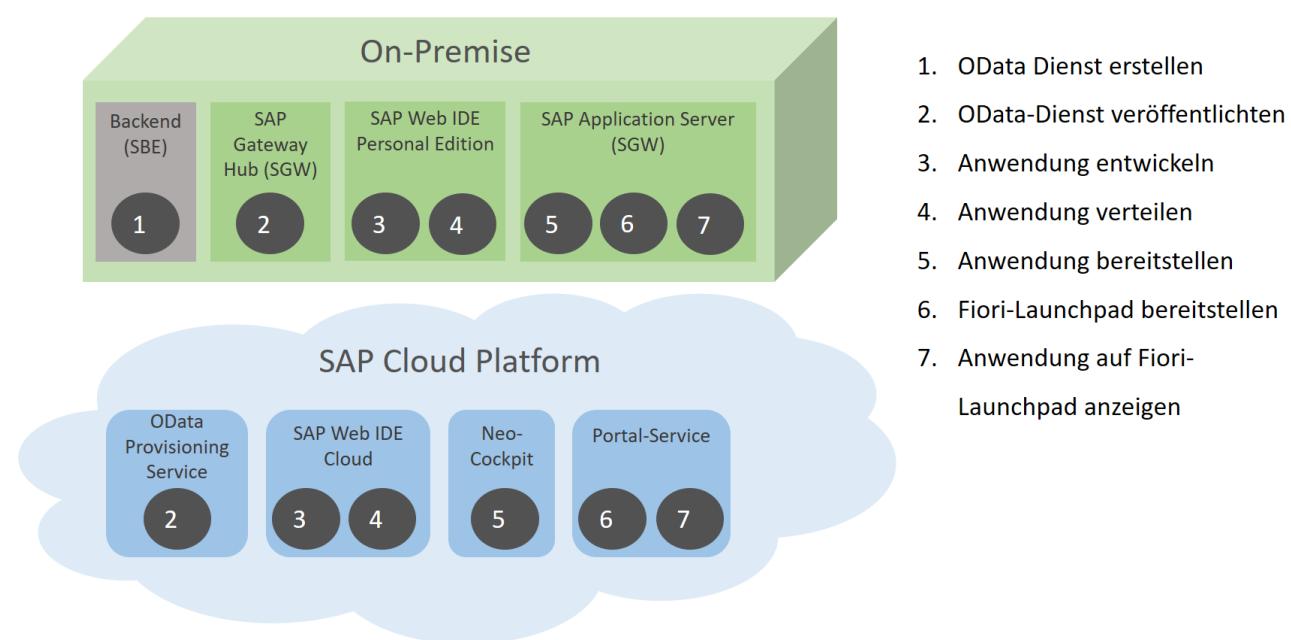


Abbildung 16: Übersicht der Umsetzung

4.1.1 Verbindung zwischen SAP Backend und SAP Cloud Platform

Wie bereits in Abschnitt 2.3.5 erwähnt, erfolgt die Anbindung von SAP-Systemen an die SAP Cloud Platform mit Hilfe des Konnektor-Dienstes [63]. Dieser kann bei den SAP Development Tools im Internet [35] heruntergeladen werden. Für die portable Version wird das Verzeichnis in einen gewünschten Ordner entpackt und die darin befindliche Datei „go.bat“ geöffnet. Die installierbare Version wird mit der Dateiendung .msi ausgeliefert und öffnet einen Assistenten zur Installation. Der SAP Cloud Connector ist nun unter der Internetadresse <https://localhost:8443/> mit der Anschlussnummer (*Port*) 8443 erreichbar, der Port kann geändert werden. Für beide Versionen wird ein *Java Development Kit* (JDK) Version 7 oder 8 benötigt. SAP bietet dies als *Java Virtual Machine* (JVM) ebenfalls bei den Development Tools an. Bei der installierten Version kann dieses im Assistenten angegeben werden. Für die portable Version sind die Umgebungsvariablen des Computers *JAVA_HOME* und Path auf das Verzeichnis des JDK zu setzen.

Parameter-name	Wert	Erläuterung
Back-end Type		Typ des lokalen Systems
	SAP Gateway	Einstellung für das SGW-System
	ABAP System	Einstellung für das SBE-System
Protocol	HTTPS	Verbindungsprotokoll
Internal Host	xxxx.perdom.intern	Internetadresse des jeweiligen Systems.
Internal Port		Anschlussnummer des lokalen Systems
	8001	Port des SGW-Systems
	5200	Port des SBE-Systems
Virtual Host	sgw/sbe	Frei wählbare Kennung unter welcher das System aus der SAP Cloud Platform erreicht werden kann
Virtual Port	8001/5200	Anschlussnummer unter welcher das System aus der SAP Cloud Platform erreicht werden kann
Principal Type	None	Wenn das verbundene System keine Authentifizierung mit Zertifikaten unterstützt.
Description		Beschreibung der Verbindung

Tabelle 1: Parameter zur Konfiguration der Verbindung im SAP Cloud Connector

Möchte man den SAP Cloud Connector gegen Ausfälle sichern, können zu einem Konto auf der SAP Cloud Platform zwei unabhängige Konnektoren zugewiesen werden, eine Master- und eine Shadow-Instanz. Der Master überträgt alle Konfigurationen zum Shadow, dieser sendet dem Master in regelmäßigen Abständen eine Abfrage und übernimmt die Master-Rolle, falls er für eine in der Dokumentation nicht näher definierte Zeitspanne keine Antwort bekommt [22 S. 320-321]. Im Rahmen dieser Arbeit werden beide Instanzen als portable Version auf demselben Rechner installiert. In der Praxis sollten sie auf unterschiedlichen Servern liegen, da sonst keine erhöhte Ausfallsicherheit gegeben ist.

Das Layout des Konnektors ist an die SAP Cloud Platform angepasst. In der Master-Instanz wird zuerst der Hochverfügbarkeitsmodus (High-Availability) aktiviert. Anschließend werden in der Shadow-Instanz die Verbindungsdaten zur Masterinstanz eingegeben und beide verbunden. Alle weiteren Konfigurationen werden in der Master-Instanz des Konnektors durchgeführt. Dort muss nun die Verbindung zu einem SAP-Cloud-Platform-Konto, in unserem Fall dem Konto aus der Neo-Umgebung, angelegt werden. Dafür befindet sich auf der Startseite ein Knopf mit der Aufschrift Unterkonto hinzufügen (*Add Subaccount*).

Anschließend muss das Konto noch mit *Connect* verbunden werden. Im Konto wird eine „Cloud To On-Premise“-Verbindung, eine Verbindung bei der die Daten im lokalen System liegen. Im umgekehrten Fall könnten die Daten auch in einer HANA-Datenbank in der SAP Cloud Platform liegen und von einer lokalen Applikation aufgerufen werden.

In der Registerkarte Zugangskontrolle (*Access Control*) muss nun das lokale System konkret zugeordnet werden. Einige Parameter haben hier mehrere Auswahlmöglichkeiten. Die Dokumentationen von SAP enthalten keine vollständige Liste aller Optionen, stattdessen sind für die meisten Nutzungsszenarien detaillierte Anleitungen vorhanden. Tabelle 1 zeigt die Einstellungen für den vorliegenden Fall. Um eine ausreichende Sicherheit zu gewährleisten, werden beide Systeme mit HTTPS und den entsprechenden Pfaden und Anschlussnummern an den SAP Cloud Connector angebunden. *Virtual Host* und *Virtual Port* sind die Bezeichnungen und Anschlussnummern, unter welchen die Systeme in der SAP Cloud Platform erreichbar sind. Dies dient ebenfalls der Sicherheit. Die tatsächlichen Pfade bleiben aus dem Internet unsichtbar. Bei dem Parameter *Principal Type* handelt es sich um eine Einstellung, die definiert, welcher Benutzer (*Principal*, siehe 11G) zur Authentifizierung bei der Einrichtung von Zielen in der SAP Cloud Platform genutzt wird [22 S. 149]. Diese Einstellung ist nur bei Nutzung von Single Sign-On relevant und erhält daher hier den Wert *Keiner(None)*.

Abschließend wird im System noch eingetragen, welche Ressourcen erreichbar sein sollen. Diese Option lässt sich auf einen oder mehrere Teilpfade beschränken. Die einfachste Lösung ist die Freigabe aller Ressourcen durch Eintragung von /sap/. Dies ist jedoch keine sichere Lösung. Daher wurden die Ressourcen auf die nötigen Pfade [64] beschränkt, wie in Abbildung 17 erkennbar. Die erste Ressource führt zum Speicherort von SAPUI5-Anwendungen, die zweite zum OData-Dienst. Beim SBE-System wird der OData-Provisioning-Dienst auf die Backend-Komponente *Information Worker - Business Enablement Provisioning* (IW-BEP) [65] zugreifen um die Liste der OData-Dienste zu erhalten. Der exponierte Pfad lautet daher /sap/iwbep/.

Resources Accessible On sgw:8001			
Enabled	Status	URL Path	Access Policy
<input checked="" type="checkbox"/>	<input type="checkbox"/>	/sap/bc/adt	Path and all sub-paths
<input checked="" type="checkbox"/>	<input type="checkbox"/>	/sap/opu/odata/sap/	Path and all sub-paths

Abbildung 17: Ressourcen auf dem SGW-System

Im SAP Cloud Platform Cockpit sind weitere Einstellungen nötig. Im Menüpunkt Konnektivität (*Connectivity*) können der Status des verbundenen Konnektors und die exponierten Backend-Systeme überprüft werden. Im gleichen Untermenü ist eine Zielverknüpfung (*Destination*) für das SGW-System anzulegen. Tabelle 2 zeigt die Parameter zur Konfiguration. Beide Verbindungen wurden über das HTTPS-Protokoll an den SAP Cloud Connector gebunden. Dieser baut einen verschlüsselten Tunnel zur SAP Cloud Platform auf. Da eine doppelte Verschlüsselung unnötig ist, wird bei der Zielkonfiguration HTTPS nicht als Option angeboten.

In großen Firmen kann es nötig sein, Systeme von unterschiedlichen Standorten, die von unterschiedlichen Administratoren verwaltet sind, mit einem Konto in der SAP Cloud Platform zu verbinden. Womöglich ist innerhalb der Firma ein Zugriff auf die Systeme auch von je nur einem Standort möglich. Damit wäre es unmöglich, alle Systeme in einen SAP Cloud Connector einzubinden. Daher wird pro Standort ein separater Konnektor verwendet und beide mit der SAP Cloud Platform verbunden. Da es theoretisch möglich ist, dass beide Konnektoren Systeme mit den exakt selben Pfaden exponieren, wird zusätzlich der Parameter *Location ID* als Kennziffer des Ortes verwendet. Zusätzlich erleichtert dies den Überblick für Nutzer und Administratoren.

Die Destination für das SBE-System wird in der Konfiguration des OData-Provisioning-Dienstes angelegt. Um den Dienst konfigurieren zu können, muss er zuerst aktiviert

werden. Dafür navigiert der Nutzer in seinem Unterkonto zu Services, sucht dort den OData-Provisioning-Dienst und klickt auf Einschalten (*Enable*). Ist dies erfolgreich kann er den Dienst konfigurieren oder öffnen. Die Konfiguration bietet die Unterseiten Ziele (*Destinations*) und Rollen (*Roles*). In *Destinations* wird das SBE-System hinterlegt, bei *Roles* finden sich zwei Rollen, welche dem Nutzer zugeordnet werden müssen, der den OData-Dienst aus dem System mit dem Cloud-Platform-Dienst veröffentlichen will.

Parametername	Wert	Beschreibung
Name	SGW-destination SBE-destination	Frei wählbarer Name der Destination
Type	HTTP	Verbindungsprotokoll
Description		Beschreibung
Location ID		Kennnummer des Cloud Konnektors
URL	http://sgw:8001 http://sbe:5200/sap/iwbep/	Internetadresse der Destination
Proxy Type	On-Premise	
Authentication	Basic Authentication	Art der Authentifizierung
Additional Properties		Zusätzliche Einstellungen
Jco.client.client	010	Für SGW
WebIDEEnabled	True	
WebIDEUsage	odata_abap,ui5_executable_abap,dev_abap,dev_gen,bsp_executable_abap	
Sap-client	100	Für SBE

Tabelle 2: Parameter zur Konfiguration von Zielen

Auch im Web-IDE-Dienst können Destinationen angelegt werden. Anders als beim OData-Provisioning-Dienst genügt dies jedoch nicht. Die Destination muss übergeordnet im Cloud Cockpit vorliegen. Bei der Web IDE Personal Edition wird eine Zielverknüpfung als Datei im Unterordner *Eclipse>config_master>service.destinations>destinations* angelegt. Diese enthält ebenfalls die Parameter aus Tabelle 2.

Tabelle 2: Parameter zur Konfiguration von Zielen

4.1.2 Einrichten eines OData-Dienstes (dessen Nutzung)

Bevor der OData-Dienst in der SAP Cloud Platform veröffentlicht und genutzt werden kann, muss er im Backend-System ausgeprägt werden. Dies erfolgt im *SAP Gateway Service Builder* welcher mit der Transaktion *SEGW* aufgerufen wird. Dort kann ein bestehendes Projekt kopiert oder ein neues Projekt angelegt werden. In Abschnitt 3.5 wurde der bereits vorhandene OData-Dienst */ASPSP/ERP_UTILITIES_UMC* erwähnt. Um diesen vor versehentlichen Änderungen zu schützen, wird er kopiert und als *YSA_ERP_UTILITIES_UMC* in dieser Arbeit behandelt.

Ein OData-Dienst besteht aus einem Datenmodell (siehe Abbildung 18) und der Implementierung des Dienstes. Im Datenmodell finden sich zuerst so genannte Entitäten. Diese sind vergleichbar mit Instanzen einer Klasse in der Objektorientierung. Sie haben Attribute und werden von Entitätstypen abgeleitet, welche selbst mit Klassen vergleichbar sind. Zwischen Entitätstypen bestehen Assoziationen, welche mit einer Kardinalität versehen sind. Kardinalität definiert, wie viele Entitäten von Typ A mit einer Entität von Typ B in Zusammenhang stehen. Die Navigation von einer Entität zur anderen erfolgt mit einem Navigationsattribut, welches auf die im Entitätstyp definierte Assoziation verweist. Mehrere Entitäten eines Typs werden als Entitätsmenge zusammengefasst.

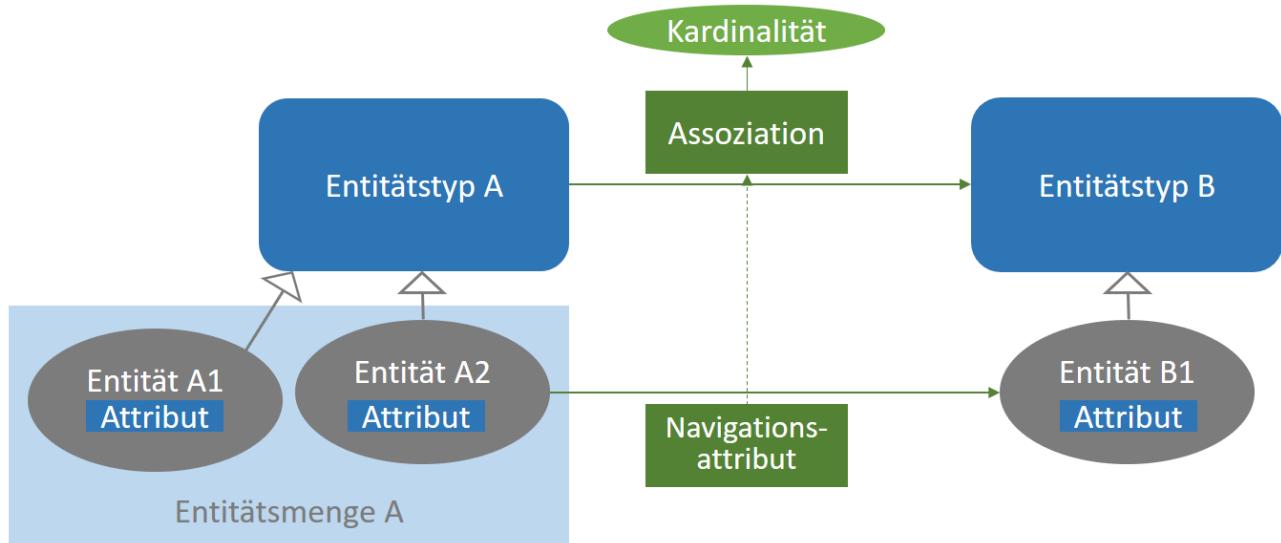


Abbildung 18: Elemente im Datenmodell eines OData-Dienstes

Der Dienst *YSA_ERP_UTILITIES_UMC* besitzt 110 Entitätstypen, für diese Anwendung relevant sind jedoch nur der Typ Geschäftspartner (*Account*) und Vertragskonto (*ContractAccount*). Wenn die Eigenschaften, welche ein Entitätstyp haben soll, mit der Beschreibung einer Tabelle im SAP-System übereinstimmen, können sie in den Gateway Service Builder importiert werden. Ansonsten ist eine manuelle Anlage auch möglich.

Zu jedem Entitätstyp wird auf Wunsch eine Entitätsmenge angelegt. Um zu beschreiben, welche Operationen der Konsument eines OData-Dienstes ausführen darf, hat SAP Annotationen eingeführt. Entitätsmengen können anlegbar, aktualisierbar, löschbar, paginierbar, adressierbar, suchbar oder abonnierbar sein. Ein Filter kann erforderlich sein.

Entitätsmengen									
	Name	Anlegbar	Aktualisierbar	Löschbar	Paginierbar	Adressierbar	Suchbar	Abonnierbar	Filter erforderlich
	Accounts		<input checked="" type="checkbox"/>						
	ContractAccounts	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>						

Abbildung 19: Annotationen der Entitätsmengen *Accounts* und *ContractAccounts*

Entitätsmengen, die nicht adressierbar sind, können in der SAP Web IDE nicht ausgewählt werden [66]. Im Fall des kopierten Dienstes sind die Entitätsmengen für den Geschäftspartner und das Vertragskonto nicht adressierbar (siehe Abbildung 19) und eine Änderung der Einstellung ist nicht möglich. Daher ist es für das vorliegende Beispiel nicht möglich diesen Dienst nutzen. Stattdessen wird ein eigener Dienst mit nur einer kleinen Menge an Eigenschaften für die Entitätstypen definiert. Zudem werden eigene Tabellen mit Testdaten erstellt.

Aus den Anforderungen an die Anwendung ergibt sich, dass die Anwendungen Informationen über Geschäftspartner und Vertragskonten benötigt. Im SAP liegen diese Informationen in den Tabellen *YSA_GP* (Geschäftspartner) und *YSA_VKGP* (Vertragskonten) in der Datenbank. Um Probleme bei Aufrufen des OData-Services in der Oberfläche durch führende Nullen zu vermeiden, wurde bei der Anlage der Tabellen darauf geachtet, dass die einge tragenen Schlüsselwerte der tatsächlichen Länge des Datenfeldes entsprechen.

Der Entitätstyp für den Geschäftspartner wird *Person* genannt. Abbildung 20 zeigt das Datenmodell des Dienstes. Die erste Entität jedes Typs enthält die Namen der Attribute, die zweite bildet einen Datensatz beispielhaft ab. Die Entitätsmengen heißen *PersonSet* und *VertragskontoSet*. Die Assoziation heißt *VertragskontoZuPerson* und nennt als Kardinalität den Wert *1:m*. Das bedeutet, dass jeder Person kein, ein oder mehrere Vertragskonten zugeordnet sein können. Primärschlüssel einer Person ist das *Partner*-Attribut. Dieses taucht auch als Fremdschlüssel in den Vertragskonten auf.

Beide Entitätsmengen werden als adressierbar annotiert. Die Eigenschaften einer Entität können folgende Annotationen erhalten: anlegbar, aktualisierbar, Nullwerte möglich, sortierbar und filterbar. Die letzten beiden Annotationen sollen für alle Eigenschaften beider Entitäten ausgewählt sein. Nullwerte sind für alle Eigenschaften erlaubt, die kein Primär- oder Fremdschlüssel sind.

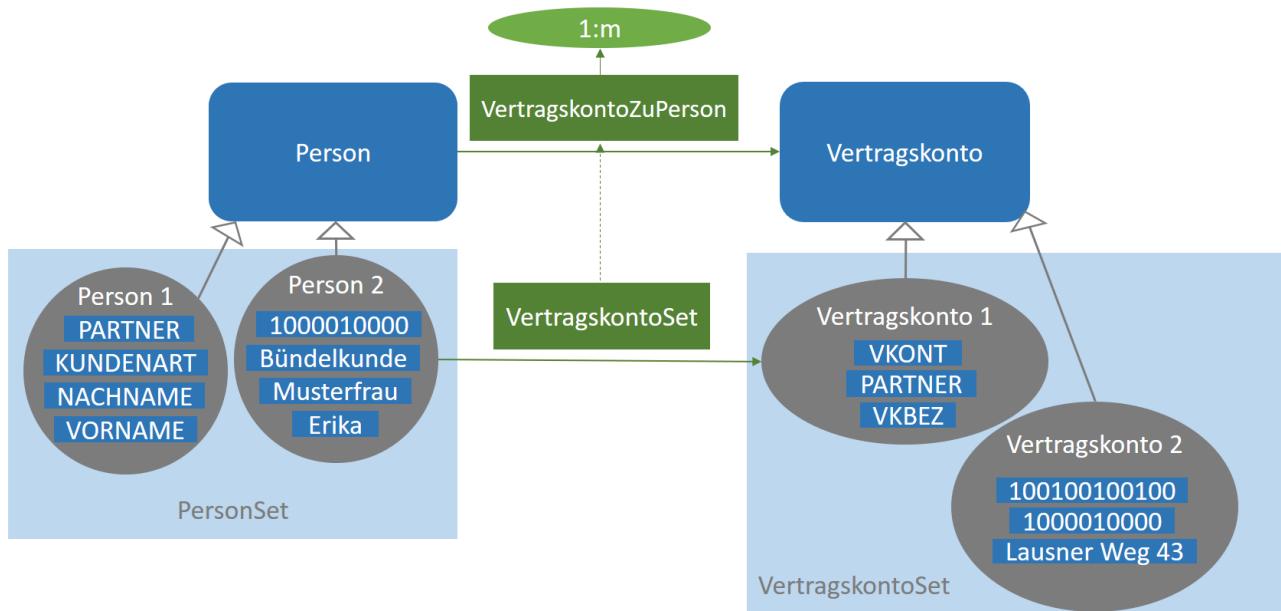


Abbildung 20: Elemente im Datenmodell des selbst definierten OData-Dienstes

In jeder Entitätsmenge im Gateway Service Builder sind automatisch die so genannten CRUD-Operationen angelegt: Erstellen (*Create*), Lesen einer Entität (*Read*, *GetEntity*), Lesen mehrerer Entitäten (*Query*, *Get EntitySet*), Ändern (*Update*) und Löschen (*Delete*). Diese Befehle lassen sich in gleicher Reihenfolge auf die bereits in Abschnitt 2.3.3 beschriebenen REST-Befehle POST, GET (für Read und Query), PUT und DELETE übertragen. Für diese Arbeit besonders relevant sind die beiden GET-Methoden, da die Daten zu den Geschäftspartnern vorerst nur angezeigt werden sollen. Folgende Abfragen werden benötigt:

- Abfrage aller Geschäftspartner
 - Filterung aller Geschäftspartner nach dem Primärschlüssel *Partner*
 - Abfrage eines einzelnen Geschäftspartners
 - Abfrage aller Vertragskonten mit einer bestimmten Partner-Nummer
 - Abfrage eines einzelnen Vertragkontos

Diese werden in der *ABAP Workbench* des SBE-Systems geändert. Der Quelltext befindet sich in Abschnitt 11H im Anhang. Anschließend werden die ausführbaren Objekte (*Laufzeitartifakte*) des Dienstes im Gateway Service Builder generiert.

Der Dienst kann nun auf zwei Arten veröffentlicht werden: über einen Zugangsserver oder mit dem OData-Provisioning-Dienst der SAP Cloud Platform. Für die erste Option wird das SGW-System in der Serviceverwaltung verknüpft und der Dienst dort registriert. Dadurch wird ihm die Adresse zugewiesen, unter dem er zukünftig in einem Browser erreichbar ist: <https://servername:8001/sap/opu/odata/sap/YSA> GEPA SRV.

The screenshot shows the 'OData Provisioning Administration' interface with a header containing a menu icon, a help icon, and a power icon. Below the header, a section titled 'Registered Services' is displayed. At the top of this section are buttons for 'Register', 'Activate/Deactivate', and 'Delete', followed by an 'Export' button. A table lists registered services with columns: Service Name, Namespace, Description, Version, Activation..., Default Destination, and Service Document. One service is listed: YSA_GEPA_SRV, which is active and has SBE-destination as its default destination. The 'Service Document' link is 'Open Service Document'.

	Service Na...	N...	Description	Version	Activation...	Default Destination	Service Document
	YSA_GEPA_SRV	SAP	Anzeigen Geschäftspartner...	1	Active	SBE-destination	Open Service Document

Abbildung 21: Hauptseite des OData-Provisioning-Dienstes

Bei der zweiten Variante wird im OData-Provisioning-Dienst auf der SAP Cloud Platform veröffentlicht. Dafür erfolgt ein Klick auf Registrieren (*Register*, siehe Abbildung 21). In einem sich öffnenden Fenster (siehe Abbildung 22) wird die für das SBE-System angelegte Zielverknüpfung ausgewählt und mit der Lupe nach OData-Diensten auf diesem System gesucht. Der passende Dienst wird durch Klick in das quadratische Feld ausgewählt ist wird mit der Registrierung automatisch auch aktiviert. Der Dienst ist jetzt in einem Webbrowser erreichbar und kann mit *Dienstdokument öffnen* (*Open Service Document*) angezeigt werden. Der Link lautet: [https://gwaas-s0019333218trial.hanatrial.ondemand.com/odata/SAP/YSA GEPA_SRV](https://gwaas-s0019333218trial.hanatrial.ondemand.com/odata/SAP/YSA_GEPA_SRV). Gwaas steht dabei für Gateway-as-a-Service.

The screenshot shows the 'Register Service' dialog. It includes a 'Select Destination' dropdown set to 'ALM-destination', a search bar labeled 'Search Services' with a magnifying glass icon, and a table listing services. The table has columns: Service Name, Service Description, Version, Service Namespace, and Model ID. Two services are listed: YSA_GEPA_SRV and YSA_SFLIGHT_PROJECT_... . The YSA_SFLIGHT_PROJECT entry is selected, indicated by a checked checkbox in the first column.

	Service Name	Service Description	Version	Service Namespace	Model ID
<input type="checkbox"/>	YSA_GEPA_SRV	Anzeigen Geschäftspartner ...	1		YSA_GEPA_MD...
<input type="checkbox"/>	YSA_SFLIGHT_PROJECT_...	Angelas Flugservice	1		YSA_SFLIGHT_...

Abbildung 22: Fenster zur Auswahl eines Dienstes

4.1.3 Erstellen und Anpassen der Oberfläche

Der OData-Dienst steht nun zur Abfrage durch eine Anwendung im Internet bereit. Für die Entwicklung der Oberfläche ist die Art der Veröffentlichung von geringer Bedeutung, solange auf die korrekte Adresse des OData-Dienstes verwiesen wird.

Bereits bei der Entwicklung der Anwendung spielt die Version des SAPUI5-Frameworks eine große Rolle. Das System, welches die Funktion anschließend bereitstellen soll, muss mit der Anwendung kompatibel sein. Soll die Anwendung auf die SAP Cloud Plattform verteilt werden, kann von Kompatibilität ausgegangen werden. Im SGW-System kann die

Version der SAPUI5-Bibliotheken unter folgender Adresse abgerufen werden: https://servername:8001/sap/public/bc/ui5_ui5/index.html Dort zeigt sich, dass das SGW SAPUI5 bis zur Version 1.28 unterstützt (siehe Abschnitt 11).

Die Anwendung soll in der SAP Web IDE der SAP Cloud Platform entwickelt werden. Dort kann ein Projekt von der SAP Cloud Platform bzw. von einem Applikationsserver importiert oder eine neue Anwendung erstellt werden. Projekte können mit einer Vorlage (*Template*), aus einem Beispielanwendung oder ohne Vorlage erstellt werden. Um eine möglichst Fiori-ähnliche Anwendung zu erstellen, bietet sich die Nutzung einer Vorlage an.

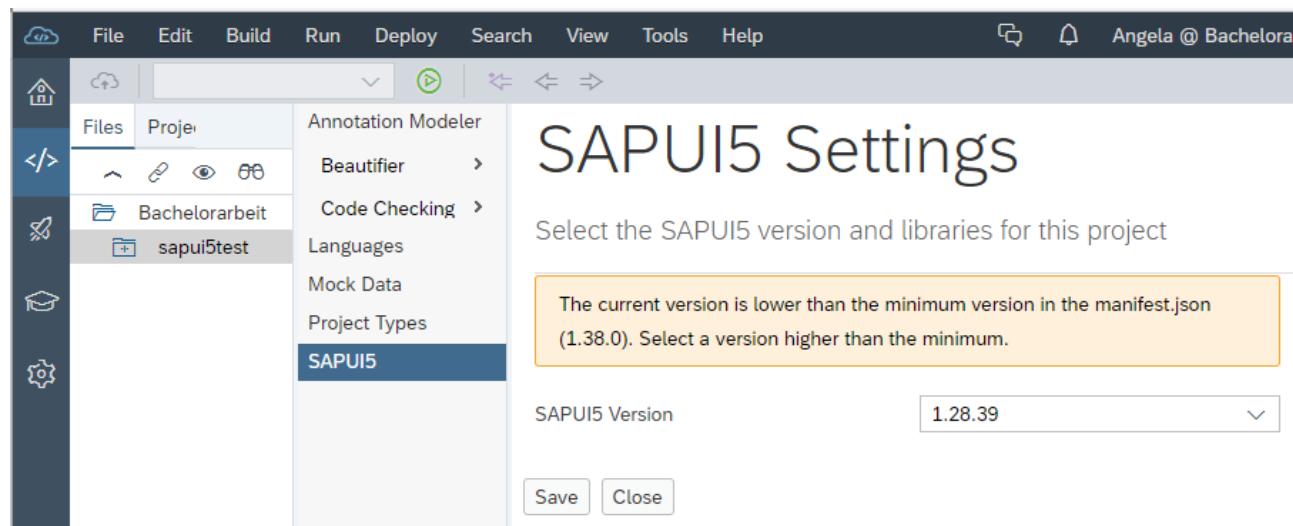


Abbildung 23: Konfiguration der SAPUI5-Version eines Projekts

Bereits hier muss auf die SAPUI5-Version geachtet werden. Die Spanne der möglichen Werte reicht von Version 1.54 bis zur Version 1.38 zurück. Keine der möglichen Vorlagen wird daher mit der Bibliothek des lokalen Applikationsservers kompatibel sein. In den Einstellungen eines Projekts ist es zwar möglich, die SAPUI5-Version zu verringern, dies führt jedoch zu einer Fehlermeldung (siehe Abbildung 23).

Anders stellt sich der Sachverhalt in der SAP Web IDE Personal Edition dar. Dort können Versionen zwischen 1.44 und 1.28 gewählt werden. Die Entwicklung einer mit dem SGW-System kompatiblen Anwendung ist damit möglich. Eine zweite Version der Anwendung wird mit der neuesten Version des SAPUI5-Frameworks in der Cloudversion der SAP Web IDE entwickelt und dokumentiert um den aktuellen Stand der Fiori-Oberflächen zu zeigen. Zu Beginn der Entwicklung hat der Nutzer die Möglichkeit die Anwendung mit oder ohne Vorlage zu entwickeln.

In der Cloudversion der SAP Web IDE stehen für die SAPUI5-Version 1.54 vierzehn Vorlagen in fünf Kategorien zur Verfügung. Die Kategorie *Fiori Elements* enthält Teilobjekte,

welche die Einhaltung der Fiori-Gestaltungsrichtlinien erleichtern sollen. Es ist möglich eigene Vorlagen zu entwickeln und diese an einige oder alle Nutzer der SAP Web IDE zu verteilen.

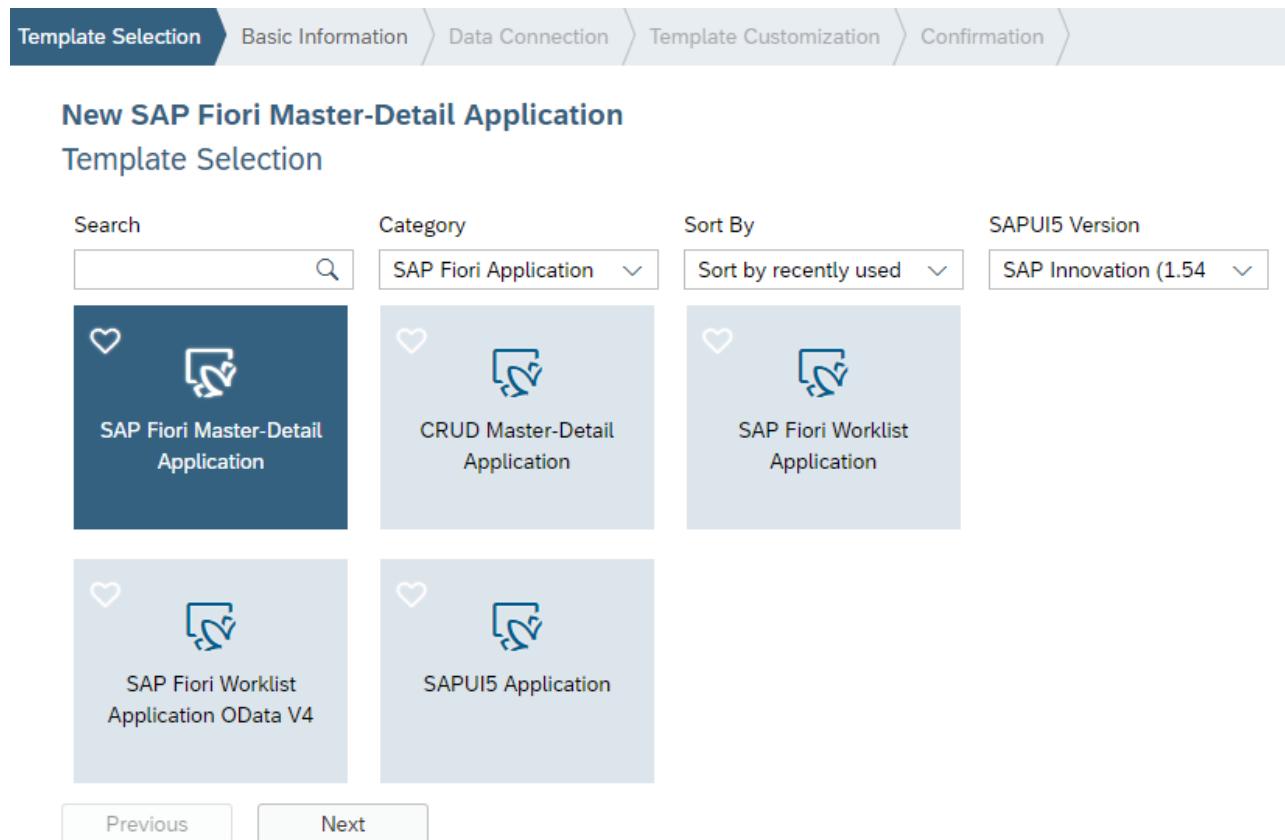


Abbildung 24: Anzeige zur Auswahl einer Projektvorlage

Vier der gegebenen Vorlagen erzeugen mit wenigen Konfigurationsschritten grundlegende Fiori-Anwendungen (siehe Abbildung 24):

- **SAP Fiori Master-Detail Application:** Die Anzeige wird in den Master- und den Detailbereich unterteilt. Der Masterbereich zeigt eine Liste von Objekten an, eines davon wird nach Auswahl im Detailbereich näher dargestellt.
- **CRUD Master-Detail Application:** Diese Vorlage bietet zusätzlich zu den Funktionen der Master-Detail-Anwendung die Möglichkeit Objekte (Datensätze) anzulegen, zu bearbeiten und zu löschen.
- **SAP Fiori Worklist Application:** Eine Liste von Objekten (Datensätzen) wird mit ausgewählten Eigenschaften (Spalten) angezeigt.
- **SAP Fiori Worklist Application OData V4:** Diese Anwendung benötigt einen OData-Dienst der Version 4 um eine Objektliste anzuzeigen.

In der Kategorie *SAP Fiori Application* wird zudem eine Vorlage für SAPUI5-Anwendungen angeboten. Diese erstellt ein Projekt mit der grundlegenden Struktur einer SAPUI5-Anwendung und einer optionalen Oberfläche. Weitere Oberflächen können manuell in der Anwendung erstellt werden.

Entwicklungsumgebung	SAP Web IDE in der SAP Cloud Platform	SAP Web IDE Personal Edition
Pfad	File>New>Project from Template	
SAPUI5-Version	1.54	1.28
Gewählte Vorlage	CRUD Master-Detail Application	SAP Fiori Master Master-Detail Application
Grundlegende Informationen	Projektname: bp_crud Titel: Partner anzeigen Namensraum: angela	Projektname: bp_mmd
Datenverbindung	Dienst-URL>GWaaS Link: /odata/SAP/YSA_GEPA_SRV Bzw. /odata/SAP/YSA_GEPA_OHNE_BUENDELKUNDE_SRV	Dienstkatalog>SGW-destination
Detaillierte Anpassung der Vorlage	Siehe Abbildung in Abschnitt 11J.3	Siehe Abbildung in Abschnitt 11J.4
Ausführen	Run>Run as>Web Application	
	flpSandbox.html	Keine Auswahl nötig

Tabelle 3: Anlegen einer Anwendung mit Hilfe einer Vorlage

Zu den in Abschnitt 3.2 definierten Anforderungen an die Anwendung passt die Vorlage für eine *CRUD-Master-Detail*-Anwendung. Die Master-Detail-Ansicht ist optisch subjektiv ansprechender als eine Worklist und die erweiterten Optionen im Vergleich zur reinen Master-Detail-Anwendung vereinfachen die zukünftige Umsetzung der Bearbeitung von Stammdaten. In der lokalen Web IDE mit geringerer SAPUI5-Version steht diese Option nicht zur Verfügung und es wird stattdessen eine *Master-Master-Detail*-Anwendung umgesetzt. Diese unterscheidet sich von der modernen Master-Detail-Anwendung dadurch, dass bei Auswahl eines Objekts aus der Masteransicht, eine weitere Masterliste mit den Unterobjekten erscheint. Der Detailbereich zeigt dann die Details des Unterobjekts an. Tabelle 3 enthält eine Übersicht der Einzelschritte zur Erzeugung der Anwendung.

In beiden Entwicklungsumgebungen öffnet sich bei der ersten Benutzung ein leerer Arbeitsbereich. Um ein Projekt aus einer Vorlage zu erstellen, wählt der Nutzer im Menü *File>New>Project from Template* und landet in dem in Abbildung 24 gezeigten Assistenten. Dort sucht er die gewünschte Vorlage aus, gibt Projektinformationen ein und verbindet die Felder mit den passenden Eigenschaften des zuvor gewählten OData-Dienstes. Bei der Anwendung in der SAP Cloud Platform stammt dieser aus dem SBE- und in der Personal Edition aus dem SGW-System. Diese Zuordnung ist willkürlich.

Die Anwendungen sind nach abgeschlossener Konfiguration der Vorlage als Webanwendung ausführbar und zeigen die Daten aus dem OData-Dienst an. Wenn der Projekttyp mit *SAP Fiori* spezifiziert wurde, kann die Anwendung auch in einer *SAP Fiori Launchpad Sandbox* getestet werden. Wird die Anwendung innerhalb eines Rahmens ausgeführt, kann zur Laufzeit zwischen allen Sprachen der Anwendungen und zwischen verschiedenen Auflösungen und Geräten gewechselt werden. Dies gibt dem Entwickler die Sicherheit, dass die Anwendung tatsächlich geräteunabhängig ist. Zur Anzeige auf einem tatsächlichen Smartphone kann in der Vorschau ein QR-Code angezeigt werden, welchen das Telefon dann scannen kann um dem Entwickler die Eingabe langer Internetadressen zu ersparen. Für die Anwendungen können Tests des Quelltextes und der Oberfläche durchgeführt werden. Einige Testszenarien sind bereits vorkonfiguriert.

An beiden Oberflächen sind noch kleine Änderungen im Quelltext nötig um die Anwendung auslieferbereit zu machen. Die Anzeige der Kundenart erfolgt in der CRUD-Anwendung im Detailbereich (siehe Abbildung 25). Dort befindet sich ein aus zwei Piktogrammen bestehendes Menü mit einem Informations- und einem Dokumentenreiter. Die Kundendetails Vorname, Name und Kundenart werden auf dem Informationsreiter in Form von Textfeldern mit Beschriftungsfeld (*Label*) hinterlegt. Dem Textfeld wird der Wert der Eigenschaft dynamisch mit der Anweisung *text={<Eigenschaft>}*, also zum Beispiel *text={Kundenart}* zugewiesen.

In den CRUD-Anwendung müssen noch einige Texte angepasst werden. Die Überschrift des Masterbereichs und die der Vertragskontoliste im Detailbereich enthalten noch

Details for Business Partner 1000010000

First Name: Erika

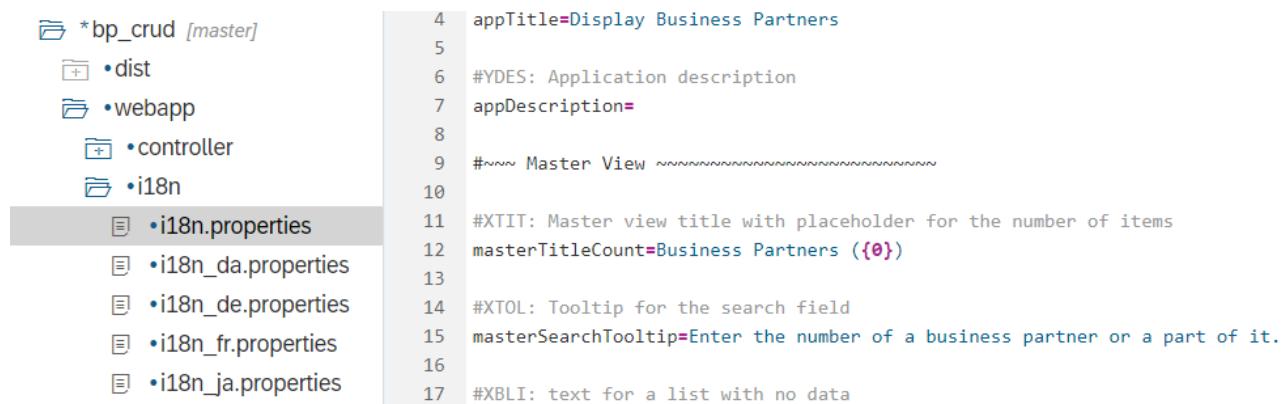
Last Name: Musterfrau

Partner Type: BÜNDELKUNDE

Abbildung 25: Anzeige der Kundenart in der CRUD-Anwendung

die Namen der Entitätsmengen (PersonSet anstatt von Geschäftspartner und Vertragskontoset anstatt von Vertragskonten, siehe Abschnitt 11K.5). Die Beschriftungstexte der Partnerdetails sollen übersetzbare gemacht werden. In der Master-Master-Detail-Anwendung sind die Bezeichnungen der Kontodetails (ebenfalls Label) noch nicht endkundentauglich.

Damit Oberflächen von Anwendungen schnell in mehreren Sprachen angezeigt werden können, bietet die Fiori-Vorlage eine Ressourcen-Datei namens „i18n.properties“ (siehe Abbildung 26). Die Abkürzung i18n steht für Internationalisierung (*Internationalisation*). Die Datei enthält Stichwort-Wert-Paare. In der Oberfläche wird an Stelle des Textes das Stichwort eingetragen und die Anwendung ermittelt zur Laufzeit die anzuseigende Sprache und die entsprechenden tatsächlichen Texte. Die verfügbaren Sprachen müssen dazu auch in den Projekteinstellungen berücksichtigt werden. Pro Sprache muss eine Ressourcendatei vorliegen.



```
4 appTitle=Display Business Partners
5
6 #YDES: Application description
7 appDescription=
8
9 #~~~ Master View ~~~~~
10
11 #XTIT: Master view title with placeholder for the number of items
12 masterTitleCount=Business Partners ({0})
13
14 #XTOL: Tooltip for the search field
15 masterSearchTooltip=Enter the number of a business partner or a part of it.
16
17 #XBLI: text for a list with no data
```

Abbildung 26: Ausschnitt aus der deutschsprachigen Ressourcen-Datei

Bei der Anpassung der Vorlage im Assistenten wurden bereits einige Stichwort-Wert-Paare gepflegt, andere sind durch SAP vorbelegt. Es ist daher sinnvoll die gegebene Datei als englische Ressourcendatei zu verstehen und weitere Texte darin ebenfalls auf Englisch zu pflegen. Die Überschrift des Masterbereichs ist mit dem Stichwort *masterTitleCount* und dem Wert *PersonSet* ({0}) hinterlegt. Die Angabe in Klammern steht für einen Platzhalter. Mithilfe einer JavaScript-Funktion wird dem Text bei Anzeige der Anwendung die tatsächliche Nummer der Geschäftspartner hinzugefügt. Dieser Wert wird zu *Business Partners* ({0}) geändert und alle anderen Werte der Datei überprüft. Für die Beschriftungstexte der Geschäftspartnerdetails werden drei weitere Stichworte benötigt. Diese sind frei wählbar, so lange sie im Quelltext der anzeigenenden Oberfläche identisch vorkommen. Um die Les- und Wartbarkeit der Anwendung durch andere Entwickler zu gewährleisten bieten sich jedoch sprechende Titel an, wie zum Beispiel *partnerNameLast* für die Beschriftung des Nachnamenfeldes. Nach diesen Änderungen wird die Anwendung in englischer Sprache wie gewünscht angezeigt.

Die Übersetzung der Oberflächen in andere Sprachen kann in der SAP Cloud Platform komfortabel aus der SAP Web IDE erfolgen. Dazu muss der Dienst *SAP Translation Hub* im SAP Cloud Platform Cockpit aktiviert und das Projekt in einem Git-Verzeichnis gespeichert sein. Im Kontextmenü der Ressourcendatei findet sich die Option Übersetzungsdateien erzeugen (*Generate Translation Files*). Diese erzeugt ein Projekt im SAP Translation Hub und startet die Erstellung der Übersetzungsdateien. Nach Ende des Prozesses erscheint ein Hinweis, dass die Dateien erstellt wurden und jetzt im Git-Verzeichnis abgefragt werden können. Die Benennung der Sprachdateien erfolgt durch Erweiterung des ausgehenden Dateinamens, aus *i18n.properties* wird zum Beispiel *i18n_de.properties*. Die Ursprungsdatei wird dabei nicht umbenannt.

Dieser Dienst steht in der SAP Web IDE Personal Edition nicht zur Verfügung. Da beide Entwicklungsumgebungen aber durch ein Git-Verzeichnis verknüpft sind (siehe Abschnitt 3.7) kann das Projekt zur Übersetzung in die SAP Cloud Platform importiert und anschließend im SAP Translation Hub übersetzt werden.

Nachdem alle Texte überprüft und gegebenenfalls überarbeitet wurden, kann die Anwendung ausgeliefert werden.

4.1.4 Verteilung (Deployment) der Anwendung

Wie in Abschnitt 3.5 beschrieben, soll die Anwendung auf zwei verschiedene Arten verteilt werden: über die SAP Cloud Platform in ein Fiori-Launchpad in der Cloud und über einen lokalen Applikationsserver in ein lokales Fiori-Launchpad. Die Cloud-Verteilung wird mit Hilfe der Cloud-Version der SAP Web IDE gezeigt, die lokale Verteilung mit der lokalen Version der Web IDE. Während die Cloudversion beide Optionen abdecken kann, ist die Verteilung einer Anwendung aus der lokalen Entwicklungsumgebung in die Cloud nicht direkt möglich. Stattdessen ist es nötig, das mit der Cloud verknüpfte Git-Verzeichnis zu aktualisieren und die Verteilung dann von dort aus durchzuführen.

Im Kontextmenü des Projekts in der Web IDE wählt der Nutzer *Deploy>Deploy to SAP Cloud Platform*. Ein sich öffnenden Fenster (siehe Abschnitt 11L) erlaubt die Verteilung einer neuen oder die Aktualisierung einer bestehenden Anwendung. Das Konto und der Projektname werden aus der Web IDE übernommen, der Anwendungsname sowie die Versionsnummer können vom Nutzer selbst vergeben werden. Mit Klick auf Verteilen („Deploy“) startet der Prozess. Nach Abschluss erscheint eine Nachricht mit der Option die Anwendung auf einem Fiori-Launchpad zu registrieren. Hierfür muss zuerst in der SAP Cloud Platform der Portal-Dienst geöffnet und dort ein Fiori-Launchpad eingerichtet werden. Auf der Startseite des Portal-Dienstes erscheint die Schaltfläche mit der eine neue Seite erstellt werden

kann. Dieser wird ein Name (*Angelas Fiori-Launchpad*) und eine Vorlage (*SAP Fiori Launchpad*) zugewiesen (siehe Abbildung 27).

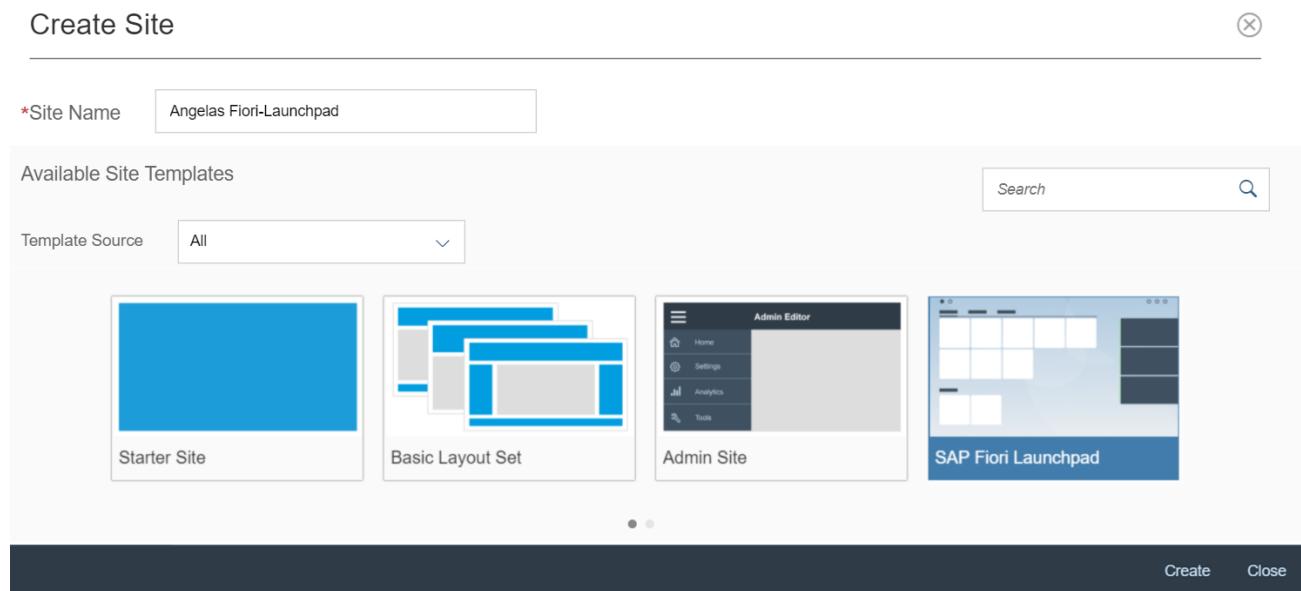


Abbildung 27: Ansicht zum Erstellen einer Seite im Portal-Dienst

Im Fiori-Launchpad muss als Nächstes ein Anwendungskatalog erstellt werden. Dieser erhält einen Namen und eine Beschreibung. Anschließend sollte ihm mindestens eine Rolle zugeordnet werden. Nutzer mit dieser Rolle haben die Berechtigung, Apps aus diesem Katalog zu sehen. Anwendungen bestehen in diesem Kontext aus Kacheln (der Oberfläche, die als Verknüpfung zur Anwendung dient) und den Zielzuordnungen (enthalten den Pfad).

Anschließend wird eine Kachelgruppe angelegt. Diese dient der übersichtlichen Organisation von Anwendungsverknüpfungen auf der Startseite des Nutzers. Sie kann Kacheln von Anwendungen aus verschiedenen Katalogen enthalten und muss ebenfalls mindestens eine Rolle zugeordnet bekommen [67]. Abbildung 28 veranschaulicht den Zusammenhang zwischen Nutzer, Rolle, Katalog, Gruppe und der Anwendung. Abbildungen zu den Einzelschritten der Umsetzung finden sich in Abschnitt 11M.

Sobald die Anwendung in der SAP Cloud Platform veröffentlicht ist, kann sie entweder über die SAP Web IDE oder den Portal-Dienst auf einem Launchpad registriert werden. Dabei müssen ein Name und eine Zielverknüpfung für das Objekt eingegeben werden. Anschließend wird eine Kachel zur Navigation erstellt und die Anwendung einem Katalog und einer Gruppe zugeordnet. Abschnitt 11N dokumentiert die nötigen Schritte in der SAP Web IDE. Bei der Anpassung der Kachel tritt ein Fehler im Assistenten auf. Die Schaltfläche für die Navigation nach vorne ist nicht auswählbar und muss durch manuelles Ändern des Kacheltyps aktiviert werden. Sobald alles konfiguriert ist, kann das Launchpad

im Portal-Dienst der SAP Cloud Platform veröffentlicht werden. Dies kann auch rückgängig gemacht werden, sobald die Anzeige nicht mehr erwünscht ist.



Abbildung 28: Zusammenspiel verschiedener Objekte im Fiori-Launchpad [68]

Die Verteilung der Anwendung auf den lokalen Applikationsserver des SGW-Systems erfolgt aus der SAP Web IDE Personal Edition. Dort führt der Weg im Kontextmenü über *Deploy>Deploy to SAPUI5 ABAP Repository*. In dem sich öffnenden Fenster (siehe Abschnitt 11O) entscheidet sich der Nutzer zuerst für ein System. Dieses muss als Destination in der lokalen Web IDE vorliegen und der Pfad `/sap/bc/adt` im SAP Cloud Connector freigegeben sein [69]. Der Assistent bestätigt, dass die Anwendung und der Applikationsserver in Bezug auf die SAPUI5-Version kompatibel sind. Mit einer Auswahl legt der Nutzer fest, ob die Anwendung neu erstellt oder eine bestehende aktualisiert werden soll. Nach erfolgreicher Verteilung erscheint ein Hinweis zur Bestätigung. Im Applikationsserver kann die Anwendung nun aktiviert und als alleinstehende Anwendung getestet werden.

Das SGW-System ist dafür ausgestattet Fiori-Launchpads zu erstellen und anzuzeigen. Zuerst erstellt der Administrator eine neues Launchpad und fügt die zuvor verteilte Anwendung hinzu. Anschließend wird ein so genanntes semantisches Objekt erstellt, welches auf die Anwendung verweist. Im Launchpad-Designer, einer Webanwendung zur Konfiguration werden anschließend analog zur Cloud-Version Katalog und Rolle erstellt. Die Anwendung muss manuell zu beiden hinzugefügt werden. Zuletzt muss im SGW-System noch eine Rolle definiert und dem Katalog, der Gruppe und dem Nutzer zugeordnet werden. Abbildungen der einzelnen Teilschritte finden sich in Abschnitt 11P. Es kann vorkommen, dass sich das Fiori Launchpad bei Änderungen nicht aktualisiert. In solchen Fällen müssen die Zwischenspeicher (*Cache*) im Applikationsserver und dem

Internetbrowser gelöscht werden [70]. Um das Launchpad in mehreren Sprachen zu testen, kann der URL-Parameter `sap-language=<zweistelliges Länderkürzel>` verwendet werden. Für die deutsche Anzeige ergibt sich somit `sap-language=de`.

4.1.5 Erreichbarkeit der Anwendung für Nutzer

Fiori-Launchpads sind in einem produktiven Umfeld viele verschiedene Rollen nötig, da es wichtig ist, zwischen den Themenbereichen zu trennen und zum Beispiel einem Buchhalter andere Ansichten zu ermöglichen als einem SAP-Berater.

Name	Type	Actions
GW_Admin	Predefined	
GW_User	Predefined	
....		

GW_User Predefined: Provisioned by the application

Individual Users	Assign	Unassign All	Groups	Assign	Unassign All
User ID		Actions	Group:		Actions

Abbildung 29: Rollenvergabe in der SAP Cloud Platform

Das lokale Fiori-Launchpad, welches auf dem SGW-System veröffentlicht wurde, bezieht seine Rollen und Benutzerdaten aus dem Applikationsserver. Ein berechtigter Nutzer erreicht die Startseite über eine feste Zielverknüpfung: <https://servername:8001/fiori>. Dort kann er sich mit seinen Benutzerdaten für das SGW-System anmelden, und sieht die ihm zugeordneten Anwendungsgruppen.

Alle Anwendungen der SAP Cloud Platform arbeiten mit einem sogenannten Identitätsanbieter, dem SAP ID Service. Dieser verwaltet Nutzer und Rollen für SAP-Anwendungen und SAP-Internetseiten. Um die Anwendung im Portal auf der SAP Cloud Platform inklusive der integrierten Daten ansehen zu können, benötigt der Nutzer die passende Berechtigungsrolle für den Portal- und den OData-Provisioning-Dienst. Wie bereits in Abschnitt 4.1.4 erwähnt, müssen dieser Rolle sowohl der Katalog als auch die Gruppe zugeordnet werden, welche die Anwendung enthalten.

Beispielhaft soll der Nutzer mit der ID *P2000134653* Zugriff zur entwickelten Anwendung im veröffentlichten Fiori-Launchpad erhalten. Für den OData-Dienst erhält er die vordefinierte Rolle *GW_User* (siehe Abbildung 29). Im Portal-Dienst bietet sich die Rolle *Portal_User* an. Möchte der Administrator die Zuweisungen nicht einzeln in der Konfiguration des Dienstes vornehmen, kann er dies auch im SAP Cloud Platform Cockpit unter dem Menüpunkt *Security>Authorizations* tun. An Stelle der vordefinierten Rollen können auch selbst definierte Rollen genutzt werden. Diese müssen in der Konfiguration der einzelnen Dienste erstellt werden.

4.1.6 Verwendung der Anwendung mit einem Standard-OData-Dienst

Die in den vorherigen Abschnitten entwickelte Anwendung ist in der Lage, eine vom Standard abweichende Datengrundlage in einem Webbrowser anzuzeigen. Weitergehend muss geprüft werden, ob die selbe Oberfläche auch die Standardversion der Datengrundlage anzeigen kann und wie groß der Änderungsaufwand ist. Dafür wird ein zweiter OData-Dienst erstellt, bei welchem das selbstentwickelte Datenfeld *Bündelkunde* nicht vorkommt. In allen anderen Punkten finden keine Änderungen statt. Der Dienst wird über den OData-Provisioning-Dienst veröffentlicht.

In der SAP Web IDE gibt es mehrere Möglichkeiten. Die Anwendung kann mit dem geänderten Dienst neu erstellt werden. Dann müssen anschließend alle Dateien aus dem bestehenden Projekt kopiert werden, die von der Vorlage abweichen, das sind die *Detail.view.xml* und die Ressourcendateien zur Übersetzung. Dabei muss auf die Anpassung aller Erwähnungen des Projektnamens im Quelltext geachtet werden.

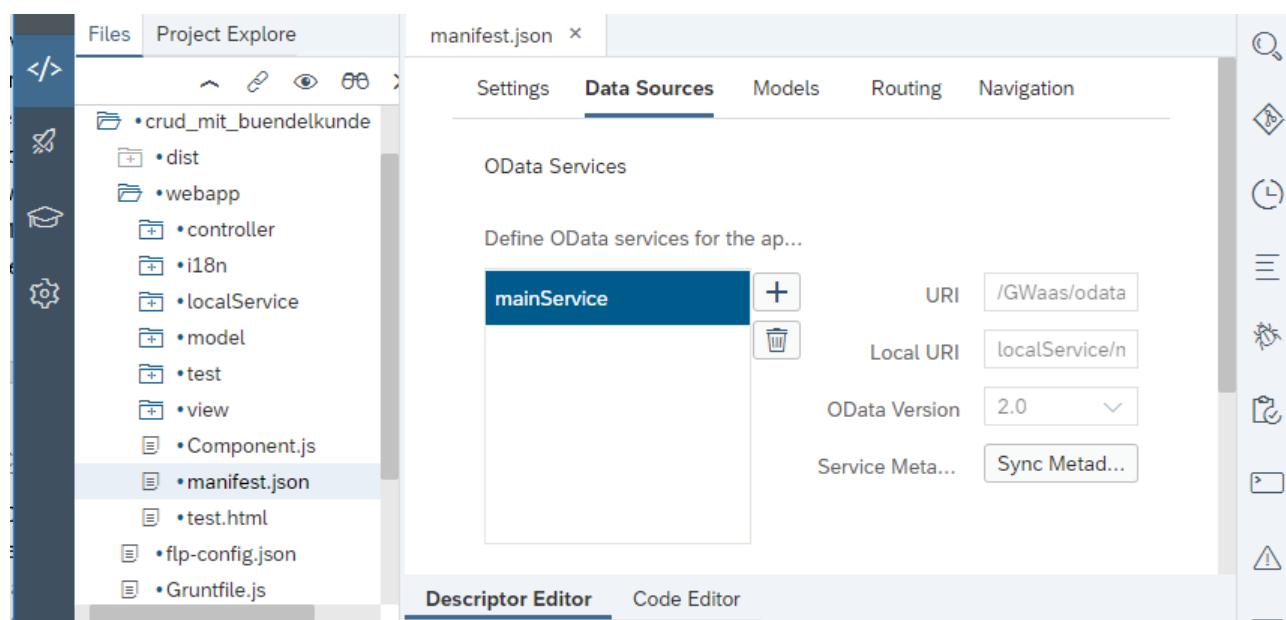


Abbildung 30: Hinzufügen eines neuen OData-Dienstes zu einem bestehenden Projekt

Die zweite Möglichkeit wäre eine Änderung des Dienstes im bestehenden Projekt. Dies kann in der *Descriptor-Editor*-Ansicht der *manifest.json*-Datei (siehe Abbildung 30) unter dem Reiter *Data Sources* erfolgen. Dort wird der Dienst analog zum Template als Dienst-URL aufgerufen. Bei einem anschließenden Test zeigt sich das selbe Verhalten, wie bei einer neuen Erstellung. Sucht man jedoch nach im Projektordner nach Referenzen zum ursprünglichen Dienst, wird schnell deutlich, dass noch Abhängigkeiten bestehen. Da sich diese auf vielfältige Weise störend auswirken können, ist die Erstellung einer *sauberen* neuen Anwendung vorzuziehen.

4.2 Einschätzung der SAP Cloud Platform

Im Rahmen der Arbeit konnte gezeigt werden, dass ein SAP-ECC-System mit Hilfe des Konnektor-Dienstes mit der SAP Cloud Platform verbunden werden kann. Wenn die Daten als OData-Dienst veröffentlicht werden, kann eine Fiori-ähnliche Anwendung darauf zugreifen und diese anzeigen. Bei der Änderung des OData-Dienstes ist in diesem Fall keine Änderung an der entwickelten Oberfläche nötig. Die Machbarkeit der gestellten Aufgabe konnte somit bestätigt werden.

Bei der Recherche und Umsetzung sind einige Punkte positiv und andere negativ aufgefallen. Da die Benutzung der Entwicklungsumgebung und die Anbindung von Systemen einen großen Teil der Arbeit ausmachen, sollen diese auch gesondert eingeschätzt werden. Zudem ist zu beachten, dass sich der Eindruck in dieser Arbeit auf die Testversion der SAP Cloud Platform bezieht. Ein Test der produktiven Version war aus finanziellen Gründen nicht möglich. Von den zwei vorhandenen Umgebungen wurde nur die Neo-Umgebung getestet.

4.2.1 Entwicklungsumgebung SAP Web IDE

Die SAP Web IDE bietet den schnellstmöglichen Start in die SAPUI5-Entwicklung. Die von SAP unterstützte Entwicklungsumgebung Eclipse kann durch Installation einer von SAP angebotenen Erweiterung vorbereitet werden. Bei WebStorm muss die Bibliothek manuell eingebunden und ein lokaler Webserver eingerichtet werden. Für Entwickler, die einen ersten Einblick in die SAPUI5-Entwicklung gewinnen wollen, ist die WebIDE daher eine gute Lösung. Ein Wechsel zu WebStorm oder einer anderen Entwicklungsumgebung kann zu einem späteren Zeitpunkt problemlos erfolgen.

Die Web IDE konnte die Entwicklung der Anwendung in diesem Fall abdecken und sich mit einigen Merkmalen positiv hervorheben. Aufgrund der vorhandenen Anwendungsvorlagen können Oberflächen besonders schnell erstellt werden. Mit Hilfe des graphischen

Layout Editors ist die Änderung von Ansichten sogar mit sehr wenigen Programmierkenntnissen möglich. Die Anbindung des SAP Translation Hub vereinfacht die Übersetzung von Anwendungen in hohem Maße, da Ressourcendateien direkt aus dem Kontextmenü übersetzt werden können.

Zum Testen der Oberfläche können zur Laufzeit Sprachen und Auflösungen verändert werden. Ein QR-Code-Generator ermöglicht die schnelle Anzeige auf einem mobilen Endgerät. Die Funktionalität des Quelltextes sichern Modultests und Oberflächentests, welche durch SAP vorkonfiguriert sind.

Die in dieser Arbeit gefundenen Nachteile der SAP Web IDE sind gegenüber den Vorteilen eher gering. Projekte im Arbeitsbereich können nicht geschlossen oder umbenannt werden. Zudem sind die Namenskonventionen für Anwendungen auf der SAP Cloud Platform unpraktisch, da sie nur Kleinbuchstaben und Zahlen zulassen. Unterstriche und Großbuchstaben, welche einen positiven Einfluss auf die Lesbarkeit von langen Namen haben, sind verboten.

Im Rahmen dieser Arbeit fiel negativ auf, dass die Anwendungsvorlagen nicht für beliebig alte SAPUI5-Versionen verfügbar waren. Mit einer niedrigeren Version entwickelte Anwendungen sind aber immerhin lauffähig. In einem produktiven Szenario sollte ohnehin auf eine aktuellere Version des SAPUI5-Frameworks zurück gegriffen werden, da diese mehr Funktionalität bietet und Fehler aus vorherigen Versionen behebt.

Die Versionsverwaltung mit Git bietet in der SAP Web IDE alle üblichen Funktionen. Die Bedienbarkeit war jedoch subjektiv betrachtet nicht immer intuitiv, einige Funktionen konnten sogar erst nach längerer Recherche in der Dokumentation der Web IDE gefunden werden.

4.2.2 Verbindungen zur Cloud Platform

Die Verbindung zwischen der SAP Cloud Platform und SAP-Backend-Systemen, die lokal installiert sind, ist dank des Konnektor-Dienstes einfach herzustellen. Im Falle einer RFC-Verbindung können aus der SAP Cloud Platform Funktionsbausteine aufgerufen werden. Dies wurde im Rahmen der Arbeit jedoch nicht getestet.

Ebenso ist eine HTTP-Verbindung möglich. Diese kann auch Backend-Systeme anderer Anbieter mit der Plattform verknüpfen. Mit dem Protokoll LDAP kann das Nutzerverzeichnis *Active Directory* von Microsoft mit der SAP Cloud Platform verbunden werden um dort als Identitätsanbieter zu dienen.

Der SAP Cloud Connector ist kostenlos und baut einen mit SSL verschlüsselten Tunnel zwischen beiden Umgebungen auf. Seine Oberfläche ist an die SAP Cloud Platform angepasst. Das trägt zur schnelleren Orientierung des Nutzers bei.

Die Anwendung steht im Internet als portable und fest-installierbare Version zur Verfügung. Letztere ist im Rahmen dieser Arbeit negativ aufgefallen, da sie sich nach Entfernung einer Java-Installation nicht mehr vollständig vom Rechner deinstallieren ließ. Aufgrund verbleibender Programmteile war anschließend eine Neuinstallation nicht mehr möglich. Dennoch bevorzugt ASP für produktive Szenarien eine Installation des Konnektors.

Um einen besonders ausfallsicheren Tunnel zu erzeugen können zwei Instanzen des Konnektor-Dienstes verbunden werden, dies nennt sich dann Hochverfügbarkeitsmodus (*High-Availability*). Die führende Instanz (*Master*) leitet alle Konfigurationen an die Hilfsinstanz (*Shadow*) weiter. Dieser prüft regelmäßig ob der Master noch erreichbar ist und übernimmt bei einem Ausfall dessen Funktion. Die Anbindung von Diensten aus dem Internet erfolgt direkt in der SAP Cloud Platform durch Anlage einer Zielverknüpfung (*Destination*).

4.2.3 SAP Cloud Plattform als Ganzes

Ein positiver Aspekt der SAP Cloud Platform ist der große Umfang der Testversion. In der Neo-Umgebung stehen vierzig Dienste zur Verfügung. Daher ist es erfreulich, dass die Testlizenz zeitlich unbeschränkt ist und keine Kosten verursacht. Die Dienste sind verzahnt und bilden vollständige Entwicklungsszenarien ab.

Die Verfügbarkeit der SAP Cloud Platform ist in der Testversion nicht garantiert. Dennoch kam es während der Erstellung dieser Arbeit nur zu einem Ausfall, welcher auf der Statusseite vermerkt und in kurzer Zeit behoben wurde. Falls die SAP Cloud Platform während einer produktiven Nutzung ausfällt und die Plattform ausschließlich zur Entwicklung von Anwendungen genutzt wird, kann mit der lokalen Version der Web IDE gearbeitet werden. Dies funktioniert natürlich nur unter der Bedingung, dass die Entwicklungsprojekte mit einem Git-Verzeichnis verknüpft und regelmäßig gespeichert wurden.

Werden die Dienste der SAP Cloud Platform zur Auslieferung von Anwendungen an Kunden genutzt, besteht ein Vertrag zwischen Arvato Systems Perdata und ihnen. Dieser schließt immer eine Dienststufenvereinbarung mit ein. Die darin garantierte Systemverfügbarkeit darf die von SAP nicht überschreiten. Ebenso dürfen die Strafen für Verstöße nicht zu weit von denen abweichen, die SAP zahlt. Wenn dies gegeben ist, entstehen für Arvato Systems Perdata bei einem Ausfall keine finanziellen Nachteile dadurch, dass der Betrieb

der Infrastruktur bei SAP liegt. Die Wartungsfenster für einzelne Dienste werden durch SAP frühzeitig angekündigt und können den Kunden rechtzeitig bekannt gegeben werden.

Zur Zielgruppe der SAP Cloud Platform gehören neben bestehenden SAP-Kunden auch Entwickler aus einem SAP-fremden Umfeld, die von der In-Memory-Technologie der HANA-Datenbank oder vom SAPUI5-Framework zur Entwicklung von modernen, geräteunabhängigen Oberflächen profitieren wollen. Diese benötigen bei Verwendung der SAP Cloud Platform keine SAP-Kenntnisse und keine weiteren SAP-Produkte. Sie können laut SAP sogar ihre eigene Programmiersprache und andere Werkzeuge in die SAP Cloud Platform integrieren. Dies wurde im Rahmen dieser Arbeit nicht überprüft.

Den hohen Ansprüchen an die Performanz von Anwendungen kommt SAP durch das Angebot verschiedener Serverstandorte (Regionen) entgegen. Die Latenzzeiten bei der Kommunikation mit Servern kann dadurch maßgeblich verringert werden. Zudem bieten die Regionen aber auch Sicherheit beim Datenschutz. Der Entwickler kann sicherstellen, dass personenbezogene Daten seiner Anwendungen, falls diese in der SAP Cloud Platform gespeichert werden, innerhalb der EU, sogar in Deutschland abgelegt werden. SAP behält sich vor, Daten in andere Länder und Regionen zu übermitteln, garantiert in diesem Fall jedoch die Einhaltung des EU-Datenschutzniveaus. Zur Sicherung der Anwendung bietet die SAP Cloud Platform moderne Technologien wie OAuth 2.0 und SAML, eigene Identitiesanbieter und eine umfangreiche Nutzer- und Rollenverwaltung.

Einige der Sicherungstechnologien ermöglichen gleichzeitig auch eine anwendungsübergreifende Anmeldung (Single Sign-On). Dies trägt zu einem positiven Nutzererlebnis bei. Innerhalb der SAP Cloud Platform und der Dienste wurde auf eine einheitliche Oberfläche geachtet, die übersichtlich sortiert ist. Die Dokumentation für alle Dienste und die Plattform an sich sind im Internet frei verfügbar. Zusätzlich gibt es zahlreiche Netztagebücher (*Blogs*) und Lernhilfen (*Tutorials*) zu einzelnen Themen der SAP Cloud Platform. Schwierig wird es nur für Nutzer, welche Schwierigkeiten mit allen der vier verfügbaren Sprachen Englisch, Chinesisch, Koreanisch und Japanisch haben.

Das Bezahlmodell der SAP Cloud Platform ist detailreich und lässt sich genau auf den Nutzer abstimmen. Genau deshalb wirkt es auch unübersichtlich und kompliziert. Ein Taschenrechner auf der Internetseite ermöglicht einen besseren Überblick. Dennoch bleiben einige Fragen offen. In dieser Arbeit wurde zum Beispiel der OData-Provisioning-Dienst verwendet, welcher laut der offiziellen Preisinformation nicht Teil der kostenlosen Testversion sein sollte. Im Falle einer produktiven Nutzung wird ein Vertrag zwischen SAP und dem Nutzer geschlossen, welcher jedoch eindeutig die vereinbarten Leistungen und Preise darstellt.

Insgesamt betrachtet, hält die SAP Cloud Platform ihre Versprechen. Sie ermöglicht einem Entwickler schnelle Erfolge in der SAPUI5-Entwicklung. Erfahrene Entwickler sehen möglicherweise eher Nachteile in einer Cloud-basierten Entwicklungsumgebung und der Nutzung eines Frameworks, weil beides nur eingeschränkt konfigurierbar ist. Nutzer mit wenig Erfahrung in der Webentwicklung können aber von der SAP Web IDE mit SAPUI5 profitieren [71].

5 Fazit und Ausblick

Im Rahmen dieser Arbeit sollte überprüft werden, ob die SAP Cloud Platform in der Lage ist energiewirtschaftliche Individualprogrammierungen aus dem System zur Planung von Unternehmensressourcen *SAP ERP Central Component (ECC)* 6.0 als *SAP-Fiori*-Anwendungen abzubilden. Hierfür wurde ein Beispiel ausgewählt und umgesetzt. Anschließend wurde geprüft, ob dessen Oberfläche nach einmaliger Implementierung für einen Energieversorger mit Standardausprägung gleichermaßen nutzbar ist.

Nach einer kurzen Erläuterung der Motivation und der Festlegung von Arbeitsthesen im ersten Kapitel beleuchtet das zweite Kapitel zuerst den theoretischen Hintergrund und erklärt in diesem Zuge die Begriffe *Cloud*, *Plattform* und *Platform-as-a-Service*. Die nächsten Abschnitte stellen die Zielgruppe, die Umgebungen und Regionen sowie die Dienste der SAP Cloud Platform vor und fassen zusammen, wie ein Nutzer Zugang zur SAP Cloud Platform bekommen kann. Abschließend werden die Möglichkeiten zur Anbindung an andere Systeme und die Verfügbarkeit und Wartung der SAP Cloud Platform festgehalten.

Der dritte Abschnitt beschäftigt sich mit den Vorüberlegungen zur Implementierung der Anwendung. Der erste Abschnitt stellt die bestehende Eigenentwicklung im SAP-ECC-System vor, der zweite leitet daraus die Anforderungen für die zu entwickelnde Anwendung ab. Anschließend werden Fiori-Anwendungen im Allgemeinen vorgestellt und geprüft, ob eine bestehende Anwendung bereits die Anforderungen erfüllt. Da dies nicht der Fall war, wird anschließend die Architektur einer selbst zu entwickelnden Anwendung festgelegt und die Themen Sicherheit und Datenschutz angesprochen. Der für die Erstellung der Anwendung besonders wichtigen Entwicklungsumgebung wurde der vorletzte Abschnitt im dritten Kapitel gewidmet, bevor es durch Auflistung der Evaluationskriterien für die SAP Cloud Platform abschließt.

Das vierte Kapitel beschreibt die einzelnen Schritte zur Umsetzung der Webanwendung: das Verbinden der lokal installierten SAP-Backend-Systeme mit der SAP Cloud Platform, das Einrichten und Veröffentlichen eines OData-Dienstes, das Erstellen und Anpassen einer Oberfläche, die Verteilung der Anwendung in der Cloud und auf einem lokalen Applikationsserver, sowie die Einbindung der veröffentlichten Anwendung auf einem Fiori-Launchpad, welches in der SAP Cloud Platform bzw. auf einem lokalen Anwendungsserver betrieben wird. Nach erfolgreicher Umsetzung und Veröffentlichung der Anwendung wurde gezeigt, dass die Anwendung, welche zur Umsetzung einer vom Standard abweichenden Daten-

grundlage konzipiert wurde, auch zur Anzeige einer im Standard vorliegenden Datengrundlage verwendet werden kann und dass dafür keine Änderungen an der Oberfläche nötig sind.

Die Machbarkeit der gestellten Aufgabe konnte im Rahmen der Arbeit gezeigt werden. Der Mehrwert der SAP Cloud Platform hängt vom genauen Nutzungsszenario und den Eigenschaften und Anforderungen des Nutzers ab. Sie eignet sich gut für bestehende SAP-Kunden, welche neue Oberflächen mit Daten aus bestehenden Backend-Systemen entwickeln wollen. Diese Anwendungen können helfen, den Kunden an neue Oberflächen zu gewöhnen und so den Umstieg vom auslaufenden SAP-ERP-ECC-6.0- auf ein S/4-HANA- System erleichtern.

5.1 Probleme

Im Rahmen dieser Arbeit traten einige Probleme auf. Zu Beginn waren diese vor allem durch die ungewohnten Begrifflichkeiten bei SAP geschuldet. Aufgrund sich ähnelnder Produktnamen konnte die Angebotslandschaft von SAP nur mühsam erfasst werden. So gibt es zum Beispiel *SAP HANA*, *SAP Business Suite 4 HANA* (*SAP S/4 HANA*), *SAP Business Suite on HANA*, *S/4 HANA Cloud* und die *SAP Cloud Platform*, welche zuvor *SAP HANA Cloud Platform* hieß.

Die SAP Cloud Platform wurde durch Arvato Systems Perdata im Vorfeld mit SAP S/4 HANA verwechselt und als nachfolgendes ERP-System verstanden. Erst im Zuge der Recherche konnte dieser Irrtum aufgeklärt und die Arbeit umstrukturiert werden. Dabei wurde zuerst darüber nachgedacht, den Titel der Arbeit zu ändern und S/4 HANA zu evaluieren. Dies scheiterte jedoch daran, dass kein S/4-HANA-System für die Erstellung dieser Arbeit zur Verfügung stand. Zudem war unsicher, ob die kostenlose Testversion des S/4-HANA-Systems in der Cloud-Version die benötigten Berechtigungen aufweisen würde um alle Aufgaben umzusetzen. In Zusammenarbeit mit dem zuständigen Mitarbeiter bei Arvato Systems Perdata wurde daher der Arbeitsinhalt so definiert, wie er in dieser Fassung vorliegt.

Eine Unsicherheit entstand durch den ständigen Wandel der SAP Cloud Platform und aller aktuellen SAP Produkte. Diese werden ständig modular weiterentwickelt und so können Erkenntnisse, welche zu Beginn der Bearbeitung gewonnen wurden, bei Abschluss der Arbeit bereits veraltet sein.

5.2 Ausblick

Diese Arbeit kann Arvato Systems als Grundlage zur Klärung weiterer Fragen dienen. In Bezug auf die Anwendung sollten zuerst das in Abschnitt 4.1.4 aufgetretenen Problem analysiert und, wenn möglich, behoben werden. Es wurde festgestellt, dass die Kacheln, welche in Fiori-Launchpads zur Darstellung der Anwendungsverknüpfungen verwendet werden, nicht dynamisch beschriftet und daher nicht übersetzt werden. Gerade auf der Übersichtsseite benötigt ein Anwender jedoch einen guten Überblick und für ihn verständliche Beschriftungen von Anwendungen. Weiterführend ist zu klären, ob die entwickelte Anwendung auch Daten aus einem S/4-HANA-Backend anzeigen kann und wie dort die Erstellung und Veröffentlichung von OData-Diensten funktioniert.

In Bezug auf das SAPUI5-Framework und die SAP Web IDE konnte in dieser Arbeit nicht untersucht werden, welcher Mehrwert für erfahrene Webentwickler besteht. Auch aus Kostengründen ist zu prüfen, ob bei der Entwicklung von Anwendungen für SAP-Systeme auf die SAP Cloud Platform verzichtet werden kann. Die integrierte Entwicklungsumgebung WebStorm der Firma JetBrains wurde bereits in Abschnitt 3.7 erwähnt und stellt eine kostengünstige Alternative zur SAP Web IDE dar. Die Schritte zur Einbindung [72] des quelloffenen OpenUI5-Frameworks und passender Werkzeugen [73] können im Internet nachvollzogen werden.

SAPUI5 selbst basiert auf anderen Oberflächen-Frameworks wie *Bootstrap* und *jQuery*. Es stellt sich daher die Frage, ob Fiori-ähnliche Anwendungen nicht auch ohne SAPUI5 mit anderen Frameworks erstellt werden können. Während die grundlegende Entwicklung damit möglicherweise mehr Zeit in Anspruch nimmt, sind die Ergebnisse anschließend immer genau auf die Anforderungen zugeschnitten und bleiben anpassbar [71].

In Bezug auf die SAP Cloud Platform ist weitergehend zu prüfen ob diese zur produktiven Auslieferung von Anwendungen genutzt werden kann. Der technische Aspekt wurde im Rahmen dieser Arbeit bereits oberflächlich geprüft, die finanzielle Komponente jedoch aus Zeitgründen vernachlässigt.

Interessant wäre auch zu prüfen, welchen Mehrwert die SAP Cloud Platform für Entwickler außerhalb des SAP-Umfelds hat. Bei spielt auch die Frage nach Konkurrenzprodukten eine Rolle.

Unabhängig von der SAP Cloud Platform wäre zudem zu überdenken, ob in Zukunft eine bessere Absicherung der SAP-Backend-Systeme durch *2-Faktor-Authentifizierung* oder die Implementierung von SSO-Methoden sinnvoll sein könnte.

6 Abkürzungsverzeichnis

ABAP	Advanced Business Application Programming
API	Application Programming Interface
App	Applikation
ASE	Adaptive Server Enterprise
ASP	Arvato Systems Perdata GmbH
CI	Customizing Include
CRUD	Create, Read, Update, Delete
CSS	Cascading Style Sheets
DSGVO	Datenschutzgrundverordnung
ECC	ERP Central Component 6.0
ERP	Enterprise Resource Planning
EU	Europäische Union
GUI	Graphical User Interface
HANA	High Performance Analytic Appliance
HTTP	Hypertext Transfer Protocol
HTTPS	HTTP Secure
I18N	Internationalisation
IaaS	Infrastructure-as-a-Service
IDE	Integrated Development Environment
IMAP	Internet Message Access Protocoll
IT	Informationstechnik
IW-BEP	Information Worker – Business Enablement Provisioning
JDK	Java Development Kit
JSON	JavaScript Object Notation
JVM	Java Virtual Machine
KDC	Key Distribution Center
LDAP	Lightweight Directory Access Protocol
NIST	National Institute of Standards and Technology
OAuth	Open Authorization
OData	Open-Data-Protokoll
PaaS	Platform-as-a-Service
POP	Post Office Protocol

QR	Quick Response
RFC	Remote Function Call
S/4 HANA	SAP Business Suite 4 HANA
SaaS	Software-as-a-Service
SAML	Security Assertion Markup Language
SAP	Systeme, Anwendungen und Produkte in der Datenverarbeitung
SAP CP	SAP Cloud Platform
SAP HANA XS	SAP HANA Extended Application Services
SAPUI5	SAP UI Development Toolkit for HTML5
SBE	SAP Backend
SE	Societas Europaea
SEGW	SAP Gateway Service Builder
SGW	SAP Gateway
SMTP	Simple Mail Transfer Protocol
SSL	Secure Sockets Layer
SSO	Single Sign-On
TCP	Transmission Control Protocol
UI	User Interface
VW	Volkswagen
WZV	Web-Zugriffsverwaltungsprodukt
XML	Extensible Markup Language

7 Abbildungsverzeichnis

Abbildung 1: Serviceebenen in der Cloud [20].....	8
Abbildung 2: SAP HANA Cloud Platform [23]	8
Abbildung 3: SAP Cloud Platform Cockpit der Neo-Umgebung.....	10
Abbildung 4: Regionen in der SAP Cloud Platform.....	11
Abbildung 5: Dienstkategorien auf der SAP Cloud Plattform [32]	12
Abbildung 6: Schematische Darstellung der Integrationsmöglichkeiten von SAP Cloud Platform [43]	16
Abbildung 7: Wartungsfenster und Upgrade-Frequenz für Cloud-Dienste.....	17
Abbildung 8: Startbild der Transaktion "BP"	19
Abbildung 9: Ansicht der Transaktion <i>BP</i>	20
Abbildung 10: Kriterien zur Umsetzung der Webanwendung	21
Abbildung 11: Bildschirmaufnahme der Filterung von SAP-Fiori-Anwendungen	23
Abbildung 12: Übersicht über die mögliche Architektur von Fiori-Applikationen	24
Abbildung 13: Anzeige der vom Client geforderten Berechtigungen bei OAuth 2.0 [77]....	27
Abbildung 14: Ablauf einer Kerberos-Authentifizierung [23 S. 305]	28
Abbildung 15: Ablauf einer SAML-Authentifizierung [57]	28
Abbildung 16: Übersicht der Umsetzung	34
Abbildung 17: Ressourcen auf dem SGW-System	37
Abbildung 18: Elemente im Datenmodell eines OData-Dienstes.....	39
Abbildung 19: Annotationen der Entitätsmengen <i>Accounts</i> und <i>ContractAccounts</i>	40
Abbildung 20: Elemente im Datenmodell des selbst definierten OData-Dienstes.....	41
Abbildung 21: Hauptseite des OData-Provisioning-Dienstes.....	42
Abbildung 22: Fenster zur Auswahl eines Dienstes.....	42
Abbildung 23: Konfiguration der SAPUI5-Version eines Projekts	43
Abbildung 24: Anzeige zur Auswahl einer Projektvorlage	44
Abbildung 25: Anzeige der Kundenart in der CRUD-Anwendung.....	46
Abbildung 26: Ausschnitt aus der deutschsprachigen Ressourcen-Datei.....	47
Abbildung 27: Ansicht zum Erstellen einer Seite im Portal-Dienst.....	49
Abbildung 28: Zusammenspiel verschiedener Objekte im Fiori-Launchpad [68]	50
Abbildung 29: Rollenvergabe in der SAP Cloud Platform	51
Abbildung 30: Hinzufügen eines neuen OData-Dienstes zu einem bestehenden Projekt .	52

8 Tabellenverzeichnis

Tabelle 1: Parameter zur Konfiguration der Verbindung im SAP Cloud Connector	35
Tabelle 2: Parameter zur Konfiguration von Zielen.....	38
Tabelle 3: Anlegen einer Anwendung mit Hilfe einer Vorlage.....	45

9 Literaturverzeichnis

- [1]. **SAP AG.** *Teilnehmerhandbuch CRM Customizing - Grundlagen.* 2010.
- [2]. **Lanninger, Volker und Wendt, Oliver.** Enzyklopädie der Wirtschaftsinformatik. *Parametrisierung von Standardsoftware.* [Online] 25. Oktober 2012. [Zitat vom: 03. April 2018.] <http://www.enzyklopaedie-der-wirtschaftsinformatik.de/lexikon/is-management/Einsatz-von-Standardanwendungssoftware/Customizing-von-Standardsoftware/Parametrisierung-von-Standardsoftware>.
- [3]. **Lanninger, Volker und Wendt, Oliver.** Enzyklopädie der Wirtschaftsinformatik. *Erweiterungsprogrammierung.* [Online] 25. Oktober 2012. [Zitat vom: 03. April 2018.] <http://www.enzyklopaedie-der-wirtschaftsinformatik.de/lexikon/is-management/Einsatz-von-Standardanwendungssoftware/Customizing-von-Standardsoftware/Erweiterungsprogrammierung>.
- [4]. **Schuh, Günther, [Hrsg.].** *Produktionsplanung und -steuerung - Grundlagen, Gestaltung und Konzepte.* 3. Auflage. Berlin Heidelberg : Springer, 2006.
- [5]. SAP Support Portal. *SAP Support Strategy.* [Online] [Zitat vom: 05. April 2018.] <https://support.sap.com/en/offering-programs/strategy.html>.
- [6]. SAP S/4 HANA. [Online] [Zitat vom: 08. Mai 2017.] <https://www.sap.com/germany/products/s4hana-erp.html>.
- [7]. **Walch, Christian.** Computerwoche. *SAP Fiori - Schneller Einstieg in die mobile IT-Strategie.* [Online] 22. Oktober 2014. [Zitat vom: 04. Mai 2018.] <https://www.computerwoche.de/a/schneller-einstieg-in-die-mobile-it-strategie,3069372,3>.
- [8]. SAP Fiori UX. [Online] März 2015. [Zitat vom: 08. Mai 2018.] <https://www.sap.com/documents/2014/06/f6a44520-5a7c-0010-82c7-eda71af511fa.html>.
- [9]. Wikipedia. *SAP Cloud Platform.* [Online] 19. Januar 2018. [Zitat vom: 08. Mai 2018.] https://de.wikipedia.org/wiki/SAP_Cloud_Platform.
- [10]. SAP SE. *Geschichte.* [Online] [Zitat vom: 07. April 2018.] <https://www.sap.com/corporate/de/company/history/>.
- [11]. Autobild. *Drei auf einer Plattform.* [Online] 05. April 2002. [Zitat vom: 11. April 2018.] <http://www.autobild.de/artikel/seat-ibiza-gegen-skoda-fabia-und-vw-polo-36151.html>.

- [12]. **Arnold, Dr. René, et al.** *Internet-basierte Plattformen und ihre Bedeutung in Deutschland*. Bad Honnef : Wissenschaftliches Institut für Infrastruktur und Kommunikationsdienste, 2016.
- [13]. **Repschläger, Jonas, Pannicke, Danny und Zarnekow, Rüdiger.** Cloud Computing: Definitionen, Geschäftsmodelle und Entwicklungspotenziale. *HMD Praxis der Wirtschaftsinformatik*. 2010, Bd. 47, 5, S. 6–15.
- [14]. **Alpar, Paul, et al.** *Anwendungsorientierte Wirtschaftsinformatik - Strategische Planung, Entwicklung und Nutzung von Informationssystemen*. Wiesbaden : Springer Vieweg, 2016.
- [15]. **Klees, Frank und Moehlmann, Thore.** Die Cloud in der digitalen Revolution und ihre Bedeutung für das SAP-Angebot. *HMD Praxis der Wirtschaftsinformatik*. 2016, Bd. 53, 5, S. 619–634.
- [16]. **Mell, Peter M. und Grance, Timothy.** *The NIST definition of cloud computing*. Gaithersburg, MD : National Institute of Standards and Technology, 2011. S. 7.
- [17]. **Baun, Christian, et al.** *Cloud Computing - Web-basierte dynamische IT-Services*. 2. Auflage, 2011.
- [18]. **Wagner, Klaus-P., Hüttl, Thomas und Backin, Dieter.** *Einführung Wirtschaftsinformatik - IT-Grundwissen für Studium und Praxis*. [Hrsg.] Iris Vieweg und Christian Werner. Wiesbaden : Gabler Verlag | Springer Fachmedien, 2012.
- [19]. **Barton, Thomas.** *E-Business mit Cloud Computing - Grundlagen, Praktische Anwendungen, verständliche Lösungsansätze*. Wiesbaden : Springer Vieweg, 2014.
- [20]. **Gilbert, Salina.** Cloudikon. *Wolkige Aussichten - Die Zeit des Cloud Computings*. [Online] 23. November 2015. [Zitat vom: 12. April 2018.] <https://cloudikon.de/wolkige-aussichten-die-zeit-des-cloud-computings/>.
- [21]. **Densborn, Frank, et al.** *Migration nach SAP S/4 HANA*. Bonn : Rheinwerk Verlag GmbH, 2017.
- [22]. SAP Help Portal. *Dokumentation der SAP Cloud Platform*. [Online] [Zitat vom: 16. April 2018.] https://help.sap.com/doc/bd6250c40c9c4c5391e3009a6f26dc3b/Cloud/en-US/SAP_Cloud_Platform.pdf.

- [23]. **Steiner, Matthias.** SAP HANA Blog. *The SAP HANA & Cloud Symbiosis*. [Online] 14. Oktober 2013. [Zitat vom: 19. April 2018.] <https://blogs.saphana.com/2013/10/14/the-sap-hana-cloud-platform-symbiosis/>.
- [24]. SAP Cloud Platform. *Success Stories*. [Online] [Zitat vom: 20. April 2018.] https://cloudplatform.sap.com/content/skywalker/website/en_us/success.html.
- [25]. **Geall, Marc.** SAP News Center. *SAP Cloud Platform and Cloud Foundry: What Does It Mean for Partners?* [Online] 11. Juli 2017. <https://news.sap.com/sap-leonardo-live-sap-cloud-platform-and-cloud-foundry-what-does-it-mean-for-partners/>.
- [26]. **Shanmugham, Murali.** GROM. *SAP Cloud Platform Goes Multi-Cloud*. [Online] 23. Juni 2017. [Zitat vom: 26. April 2018.] <http://www.grom.com/sap-cloud-platform-goes-multi-cloud/>.
- [27]. **Neumann, Alexander.** Heise Developer. *PaaS: Zertifizierungsprogramm für Cloud Foundry gestartet*. [Online] 17. Dezember 2015. [Zitat vom: 23. April 2018.] <https://www.heise.de/developer/meldung/PaaS-Zertifizierungsprogramm-fuer-Cloud-Foundry-gestartet-3045722.html>.
- [28]. **Ramamoorthy, Hariprasauth.** SAP Community. *Getting started with Python development – Bring Your Own Language*. [Online] 17. Mai 2017. [Zitat vom: 23. April 2018.] <https://blogs.sap.com/2017/05/17/getting-started-with-python-development-bring-your-own-language>.
- [29]. **Neumann, Alexander.** Heise Developer. *NetWeaver Cloud – Platform as a Service à la SAP*. [Online] 21. Juni 2012. [Zitat vom: 23. April 2018.] <https://www.heise.de/developer/meldung/NetWeaver-Cloud-Platform-as-a-Service-a-la-SAP-1622874.html>.
- [30]. **Kazi, Aiaz.** SAP HANA Blog. *Evolution of the SAP HANA Cloud Platform*. [Online] 10. Mai 2013. [Zitat vom: 23. April 2018.] <https://blogs.saphana.com/2013/05/10/evolution-of-the-sap-hana-cloud-platform>.
- [31]. Chip. *Die Plattform: SAP NetWeaver Cloud*. [Online] 08. Oktober 2012. [Zitat vom: 23. April 2018.] http://www.chip.de/artikel/Artikelserie-Cloud-Computing-SAP-4_57786088.html.
- [32]. SAP Cloud Platform. *Capabilities*. [Online] [Zitat vom: 20. April 2018.] <https://cloudplatform.sap.com/capabilities.html>.

- [33]. Wikipedia. *DevOps*. [Online] 15. Juni 2018. [Zitat vom: 15. Juni 2018.] <https://de.wikipedia.org/wiki/DevOps>.
- [34]. **Keidar, Michal.** SAP Blog. *SAP Web IDE Versions... Which do I use...?* [Online] 19. September 2017. [Zitat vom: 28. Mai 2018.] <https://blogs.sap.com/2017/09/19/sap-web-ide-versions...-which-do-i-use.../>.
- [35]. SAP Development Tools. [Online] [Zitat vom: 28. Mai 2018.] <https://tools.hana.ondemand.com>.
- [36]. SAP Help Portal. *SAP Web IDE Full-Stack*. [Online] 23. Mai 2018. [Zitat vom: 28. Mai 2018.] https://help.sap.com/doc/97abfdee97f744ed8987154fe3a78c0f/CF/en-US/SAP_Web_IDE_en.pdf.
- [37]. SAP Help Portal. *SAP Translation Hub*. [Online] [Zitat vom: 15. Juni 2018.] <https://help.sap.com/viewer/ed6ce7a29bdd42169f5f0d7868bce6eb/Cloud/en-US/1b15cf69580449c0bd8525696c97b90d.html>.
- [38]. **Bönnen, Carsten, et al.** *OData und SAP Gateway*. Bonn : Galileo Press, 2014.
- [39]. **Plessner, Christoph.** Computerwoche. *Self-Service-Portale in CRM integrieren*. [Online] 09. Juli 2013. [Zitat vom: 07. Mai 2018.] <https://www.computerwoche.de/a/self-service-portale-in-crm-integrieren,2541839>.
- [40]. SAP Cloud Platform. *Pricing*. [Online] [Zitat vom: 20. April 2017.] <https://cloudplatform.sap.com/pricing.html>.
- [41]. SAP Cloud Platform. *Pricing and Packages*. [Online] 2017. [Zitat vom: 20. April 2018.] https://cloudplatform.sap.com/content/dam/website/skywalker/en_us/PDFs/SAP_CP_Pricing_PDF_2_21.pdf.
- [42]. **Krämer, Hr.** Support-Chat auf SAP.de. 2018.
- [43]. [Online] <https://www.slideshare.net/IBSolutionGmbH/integration-von-cloud-und-onpremise-mit-der-sap-hana-cloud-platform>.
- [44]. SAP Cloud Trust Center. *Cloud Services*. [Online] [Zitat vom: 15. Juni 2018.] <https://www.sap.com/about/cloud-trust-center/cloud-service-level-agreements/cloud-services.html>.

- [45]. RheinEnergie. *Energielexikon*. [Online] [Zitat vom: 06. Mai 2018.] https://www.rheinenergie.com/de/privatkundenportal/service_1/energielexikon/index.php?letter=B.
- [46]. **Knabe, Prof. Ch.** Beuth Hochschule für Technik Berlin. *Pflichtenheft / Lastenheft*. [Online] 25. 09. 2006. [Zitat vom: 25. 05. 2018.] <http://public.beuth-hochschule.de/~knabe/fach/swp-i/Definition-PH.pdf>.
- [47]. **Englbrecht, Michael und Wegelin, Michael.** *SAP Fiori - Implementierung und Entwicklung*. 2., aktualisierte und erweiterte Auflage. Bonn : Rheinwerk Verlag GmbH, 2017.
- [48]. **Gorshkova, E.A., et al.** A UML-Based Modeling of Web Application Controller. *Programming and Computer Software*. 2005, Bd. Vol. 31, Nr. 1, S. 29-33.
- [49]. **Kunz, Andreas.** SAP Blog. *What is OpenUI5 / SAPUI5 ?* [Online] 11. Dezember 2013. [Zitat vom: 15. Juni 2018.] <https://blogs.sap.com/2013/12/11/what-is-openui5-sapui5>.
- [50]. Fiori Design Guidelines. [Online] [Zitat vom: 12. Juni 2018.] <https://experience.sap.com/fiori-design/>.
- [51]. **Taté Tsiledze, Jean-Francois.** Conet Technologie-Blog. *Der Weg zur ersten Fiori App*. [Online] 01. März 2016. [Zitat vom: 08. Mai 2018.] <https://www.conet.de/blog/technologien/sap/2016-03-01/weg-zur-ersten-fiori-app-3422>.
- [52]. Fiori App Reference Library. *Display Business Partners*. [Online] [Zitat vom: 07. Mai 2018.] [https://fioriappslibrary.hana.ondemand.com/sap/fix/externalViewer/#/detail/Apps\('BUP3'\)/S100P](https://fioriappslibrary.hana.ondemand.com/sap/fix/externalViewer/#/detail/Apps('BUP3')/S100P).
- [53]. SAP Documentation. *Business Enablement Provisioning (IW_BEP)*. [Online] 15. Februar 2018. [Zitat vom: 22. Mai 2018.] https://help.sap.com/saphelp_gateway20sp12/helpdata/en/a2/15f0b42f2948f6bb9e51f98e8c39e9/frameset.htm.
- [54]. **Fischer, Andre.** SAP Blog. *SAP Gateway deployment options in a nutshell*. [Online] 27. Mai 2013. [Zitat vom: 08. Mai 2018.] <https://blogs.sap.com/2013/05/27/sap-netweaver-gateway-deployment-options-in-a-nutshell/>.

[55]. SAP Help Portal. *SAP Fiori: Einrichtung und Konfiguration*. [Online] [Zitat vom: 01. Juni 2018.] <https://help.sap.com/viewer/d71464d9f3204ea8be1144d62acd9ac3/7.52/de-DE/270dd0bc79044592ab22545227f2435b.html>.

[56]. SAP Dokumentation. *Single Sign-On für Web-basierten Zugriff*. [Online] [Zitat vom: 03. Juni 2018.] https://help.sap.com/saphelp_nw73ehp1/helpdata/de/4f/991d85b10c16c7e10000000a42189d/frameset.htm.

[57]. SAP Help Portal. *Identity Provider for Single Sign-On and SAP Identity Management*. [Online] 15. Mai 2017. [Zitat vom: 04. Juni 2018.] https://help.sap.com/doc/339459818c4e4cb881c353e04a037a97/2.15/en-US/IdentityProviderForSAPSsingleSign-OnAndSAPIidentityManagement_uacp.pdf.

[58]. **Bager, Jo.** Hinter die Mauern - Warum Sie im Jahr 2017 Ihre Daten in der EU behalten sollten. c't. 2017, 09/2007, S. 70.

[59]. SAP-Datenschutzerklärung. [Online] 22. Januar 2018. [Zitat vom: 31. Mai 2018.] <https://www.sap.com/germany/about/legal/privacy.html>.

[60]. Terms of Use Agreement for the SAP Cloud Platform Site. [Online] [Zitat vom: 31. Mai 2018.] https://help.hana.ondemand.com/terms_of_use.html.

[61]. **Schäfer, Holger.** OpenUI Developer. *The Swiss Knife for UI5 Developers*. [Online] 2014. [Zitat vom: 15. Juni 2018.] <https://openui5.blogspot.com/p/ide.html>.

[62]. SAP Blog. *SAP Web IDE – Online and Offline development with Web IDE on HCP and Web IDE Personal Edition*. [Online] 05. Oktober 2016. [Zitat vom: 01. Juni 2018.] <https://blogs.sap.com/2016/10/05/how-to-use-the-hana-cloud-platform-web-ide-personal-edition/>.

[63]. SAP Blog. *Connect ABAP Backend to HCP via HCP OData provisioning*. [Online] 14. Oktober 2016. [Zitat vom: 05. Juni 2018.] <https://blogs.sap.com/2016/10/14/connect-apab-backend-hcp-via-hci-odata-provisioning/>.

[64]. SAP Help Portal. *Connect to ABAP Systems*. [Online] [Zitat vom: 21. Juni 2018.] <https://help.sap.com/viewer/825270ffffe74d9f988a0f0066ad59f0/Cloud/en-US/5c3debce758a470e8342161457fd6f70.html>.

- [65]. SAP Archive. *IWFND and IWBEPI*. [Online] [Zitat vom: 05. Juni 2018.] <https://archive.sap.com/discussions/thread/3570063>.
- [66]. SAP Community WIKI. *SAP Web IDE - Technical FAQ*. [Online] 04. April 2018. [Zitat vom: 06. Juni 2018.] <https://wiki.scn.sap.com/wiki/display/SWI/SAP+Web+IDE+-+Technical+FAQ>.
- [67]. SAP Help Portal. *SAP Cloud Platform Portal*. [Online] [Zitat vom: 12. Juni 2018.] <https://help.sap.com/viewer/3ca6847da92847d79b27753d690ac5d5/Cloud/en-US/18b3fc15e6fe487baa99be07c1da82c3.html>.
- [68]. SAP HANA Cloud Documentation. *About SAP Fiori Launchpad Objects*. [Online] [Zitat vom: 14. Juni 2018.] https://help.hana.ondemand.com/cloud_portal_flp/frameset.htm?18b3fc15e6fe487baa99be07c1da82c3.html.
- [69]. **Keidar, Michal.** SAP Blog. *Troubleshooting Deployment to an ABAP System from SAP Web IDE*. [Online] 25. Februar 2016. [Zitat vom: 13. Juni 2018.] <https://blogs.sap.com/2016/02/25/troubleshooting-deployment-to-abap-system-from-sap-web-ide/>.
- [70]. SAP Blog. *Cache Maintenance in Fiori*. [Online] 02. März 2016. [Zitat vom: 13. Juni 2018.] <https://blogs.sap.com/2016/03/02/cache-maintenance-in-fiori>.
- [71]. **Hirschmann, Dirk und Mistler, Elias.** HTML5 vs. SAPUI5. [Online] [Zitat vom: 19. Juni 2018.] https://www.pikon.com/fileadmin/Dateien/PIKON_DE/Artikel-pdfs/HTML5_vs_SAPUI5_E3.pdf.
- [72]. **van het Hof, Robin.** Sap Blog. *Configuring JetBrains WebStorm for UI5 development*. [Online] 22. September 2014. [Zitat vom: 19. Juni 2018.] <https://blogs.sap.com/2014/09/22/configuring-jetbrains-webstorm-for-ui5-development/>.
- [73]. **Schuff, Christian.** SAP Blog. *UI5ers Buzz #18: Useful Little Helpers—Faster SAPUI5 Development in WebStorm*. [Online] 20. November 2017. [Zitat vom: 19. Juni 2018.] <https://blogs.sap.com/2017/11/20/ui5ers-buzz-18-useful-little-helpers%E2%80%8A-%E2%80%8Afaster-sapui5-development-in-webstorm/>.
- [74]. Wikipedia. *Cloud Computing*. [Online] [Zitat vom: 12. April 2018.] https://de.wikipedia.org/wiki/Cloud_Computing#/media/File:Cloud_computing.svg.

[75]. SAP Help Portal. *SAP Cloud Platform regions and service portfolio*. [Online] [Zitat vom: 26. April 2018.] <https://help.sap.com/doc/aa1ccd10da6c4337aa737df2ead1855b/Cloud/en-US/3b642f68227b4b1398d2ce1a5351389a.html>.

[76]. **Plattner, Prof. Dr. Hasso.** Enzyklopädie der Wirtschaftsinformatik. *In-Memory Data Management*. [Online] 05. September 2017. [Zitat vom: 02. Mai 2018.] <http://www.enzyklopaedie-der-wirtschaftsinformatik.de/lexikon/datenwissen/Datenmanagement/Datenbanksystem/In-Memory-Data-Management/index.html/?searchterm=in-memory>.

[77]. **Noyes, Katherine.** PCWorld. *SAP unwraps a new enterprise suite based on Hana*. [Online] 03. Februar 2015. [Zitat vom: 19. April 2018.] <https://www.pcworld.com/article/2879512/sap-unwraps-a-new-enterprise-suite-based-on-hana.html>.

[78]. SAP ERP. [Online] 07. Februar 2018. [Zitat vom: 10. April 2018.] https://de.wikipedia.org/w/index.php?title=SAP_ERP&oldid=173743809.

[79]. **Fehling, Christoph und Leymann, Prof. Dr. Frank.** Gabler Wirtschaftslexikon. *Cloud Computing*. [Online] 20. Februar 2018. [Zitat vom: 11. April 2018.] <https://wirtschaftslexikon.gabler.de/definition/cloud-computing-53360/version-276453>.

[80]. SAP S/4 HANA Cloud. [Online] [Zitat vom: 09. April 2018.] <https://www.sap.com/germany/products/s4hana-erp-cloud.html>.

[81]. Stanoevska_Grid and Cloud Computing.

[82]. **Hahn, Christopher.** Digitalisierung der IT-Industrie mit Cloud Plattformen – Implikationen für Entwickler und Anwender. *HMD Praxis der Wirtschaftsinformatik*. 2016, Bd. 53, 5, S. 594–606.

[83]. **Hentschel, Raoul und Leyh, Christian.** Cloud Computing: Gestern, heute, morgen. *HMD Praxis der Wirtschaftsinformatik*. 2016, Bd. 53, 5, S. 563–579.

[84]. **Knoll, Matthias und Rinderle-Ma, Stefanie.** Plattformen – Eine Einführung. *HMD Praxis der Wirtschaftsinformatik*. 2015, Bd. 52, 3, S. 322–336.

[85]. SAP App Center. [Online] [Zitat vom: 20. April 2018.] <https://www.sapappcenter.com/home>.

[86]. Datenschutzbeauftragter Info. *Räumlicher Anwendungsbereich: Wo gilt die DSGVO?* [Online] 15. Mai 2017. [Zitat vom: 25. April 2018.] <https://www.datenschutzbeauftragter-info.de/raeumlicher-anwendungsbereich-wo-gilt-die-dsgvo/>.

[87]. **Voigt, Paul und von dem Bussche, Axel.** *EU-Datenschutz-Grundverordnung (DSGVO)*. s.l. : Springer Verlag, 2018.

[88]. Wikipedia. *Integration (Software)*. [Online] 03. April 2013. [Zitat vom: 29. April 2018.] [https://de.wikipedia.org/wiki/Integration_\(Software\)](https://de.wikipedia.org/wiki/Integration_(Software)).

[89]. SAP Cloud Platform Connectivity. *Cloud Connector*. [Online] [Zitat vom: 01. Mai 2018.] <https://help.sap.com/viewer/cca91383641e40ffbe03bcd78f00f681/Cloud/en-US/e6c7616abb5710148cf3e75d96d596.html>.

[90]. **Shanmugham, Murali.** Fiori Cloud and supported landscape scenarios. [Online] 06. Juli 2017. [Zitat vom: 03. Mai 2018.] <https://blogs.sap.com/2017/07/06/fiori-cloud-and-supported-landscape-scenarios/>.

[91]. Wikipedia. *Europäische Gesellschaft*. [Online] 17. April 2018. [Zitat vom: 08. Mai 2018.] https://de.wikipedia.org/wiki/Europ%C3%A4ische_Gesellschaft.

[92]. **Manechini, Cezar.** SAP Blog. *Connecting SAP Fiori Cloud App Approve Prochase Orders to On-Premise Landscape - Part I.* [Online] 18. Oktober 2017. [Zitat vom: 08. Mai 2018.] <https://blogs.sap.com/2017/10/18/connecting-sap-fiori-cloud-app-approve-purchase-orders-to-on-premise-landscape-part-i/>.

[93]. SAP Help Portal. *Connecting a Customer System to SAP Cloud Platform Integration*. [Online] 15. September 2017. [Zitat vom: 30. Mai 2018.] https://help.sap.com/doc/61b8bd715cf94d9e8234b6e1480aeb53/Cloud/en-US/SAP_HCI_OnboardingGuide_External.pdf.

[94]. SAP Help Portal. *Authentifizierung mit X.509-Client-Zertifikaten*. [Online] [Zitat vom: 03. 06 2018.] https://help.sap.com/doc/saphelp_nw70ehp2/7.02.16/de-DE/b1/07dd3aeedb7445e10000000a114084/content.htm?no_cache=true.

[95]. OAuth2 Tutorial: Google as Authentication Service for Web Applications. [Online] [Zitat vom: 05. Juni 2018.] <https://www.membrane-soa.org/service-proxy-doc/4.4/oauth2-google.htm>.

[96]. SAP Dokumentation. *Einrichtung von Katalogen, Gruppen und Rollen im SAP Fiori Launchpad.* [Online] [Zitat vom: 12. Juni 2018.]

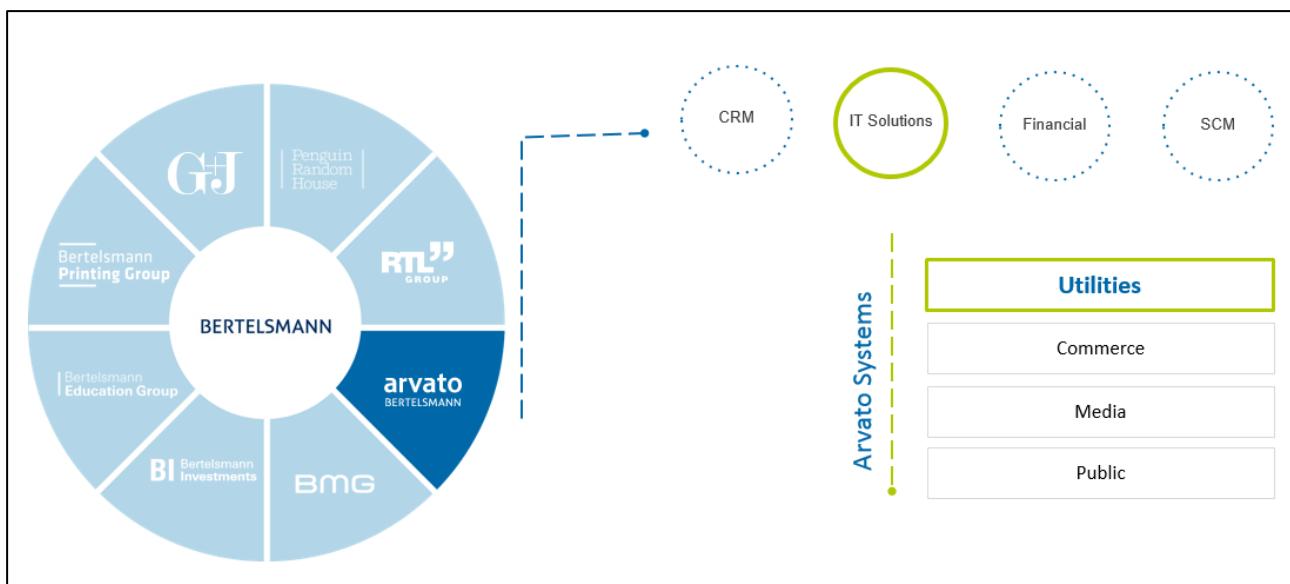
https://help.sap.com/doc/saphelp_ssbs/1.0/de-DE/6b/e47c525ae17154e10000000a44176d/frameset.htm.

10 Anhangsverzeichnis

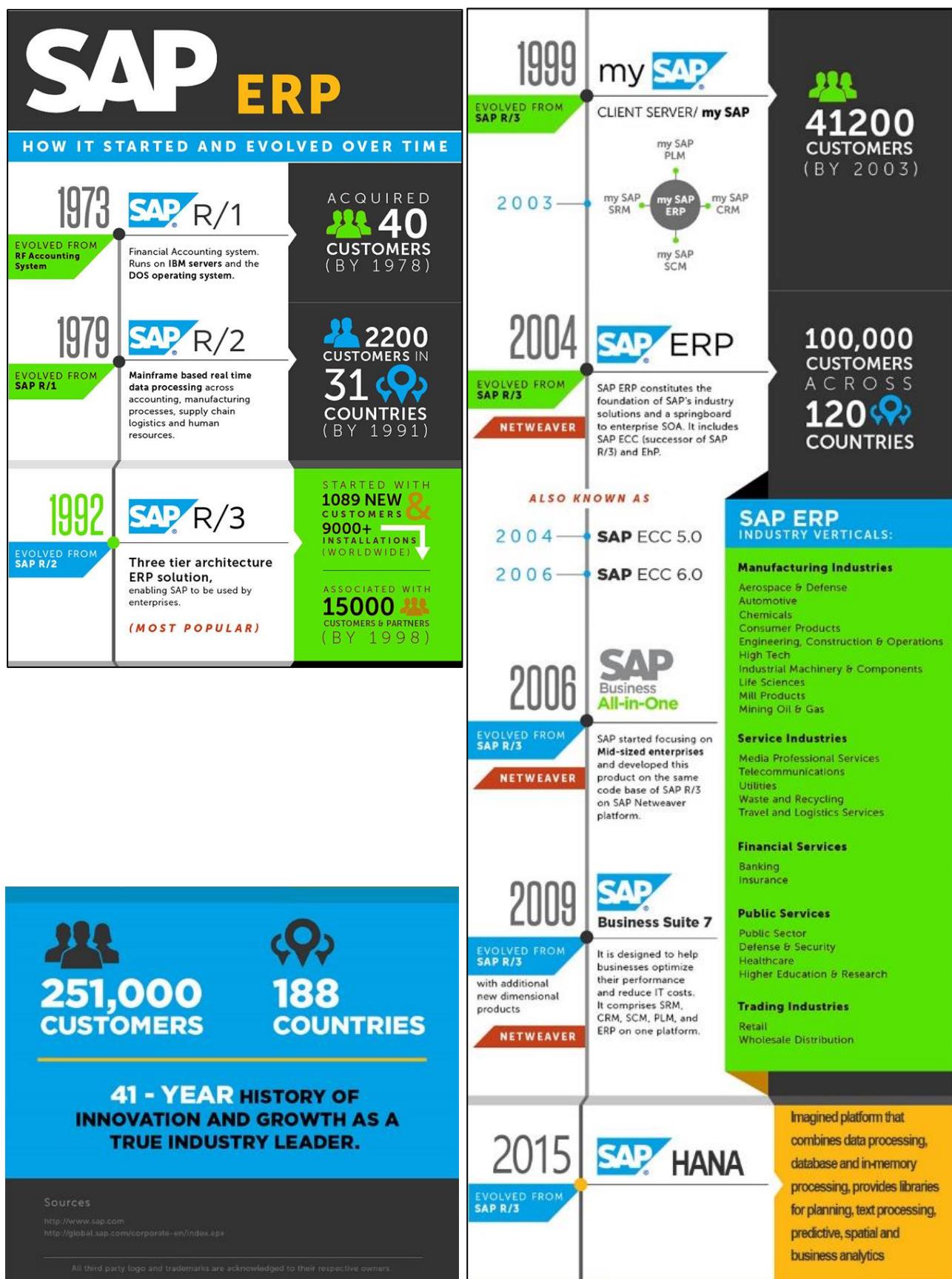
A	Übersicht über die Konzernstruktur	XIX
B	Versionen des SAP ERP	XX
C	Überblick über Cloud Computing [74]	XXI
D	SAP Support-Chat	XXII
E	Überblick über Testkonten in der SAP Cloud Platform	XXIII
F	Rechenzentren als Standort für SAP Cloud Platform [75]	XXIV
G	Definition des Wortes <i>Principal</i> in Transaktion SAPTERM	XXV
H	Quelltexte zur Implementierung des OData-Dienstes.....	XXVI
I	Informationen zur Version des SAPUI5-Frameworks im SGW-System	XXVIII
J	Anpassen der Anwendungsvorlage	XXIX
K	Ansicht der Anwendung nach der Erstellung	XXXI
L	Verteilen einer Anwendung in der SAP Cloud Platform	XXXIII
M	Anlegen eines Fiori-Launchpad im Portal-Dienst der SAP Cloud Platform.....	XXXIV
N	Registrierung einer Anwendung auf einem Launchpad in der SAP Cloud Platform	
	XXXVI	
O	Verteilen einer Anwendung auf einem lokalen Applikationsserver.....	XXXVIII
P	Registrieren einer Anwendung auf einem lokalen Fiori-Launchpad.....	XL

11 Anhang

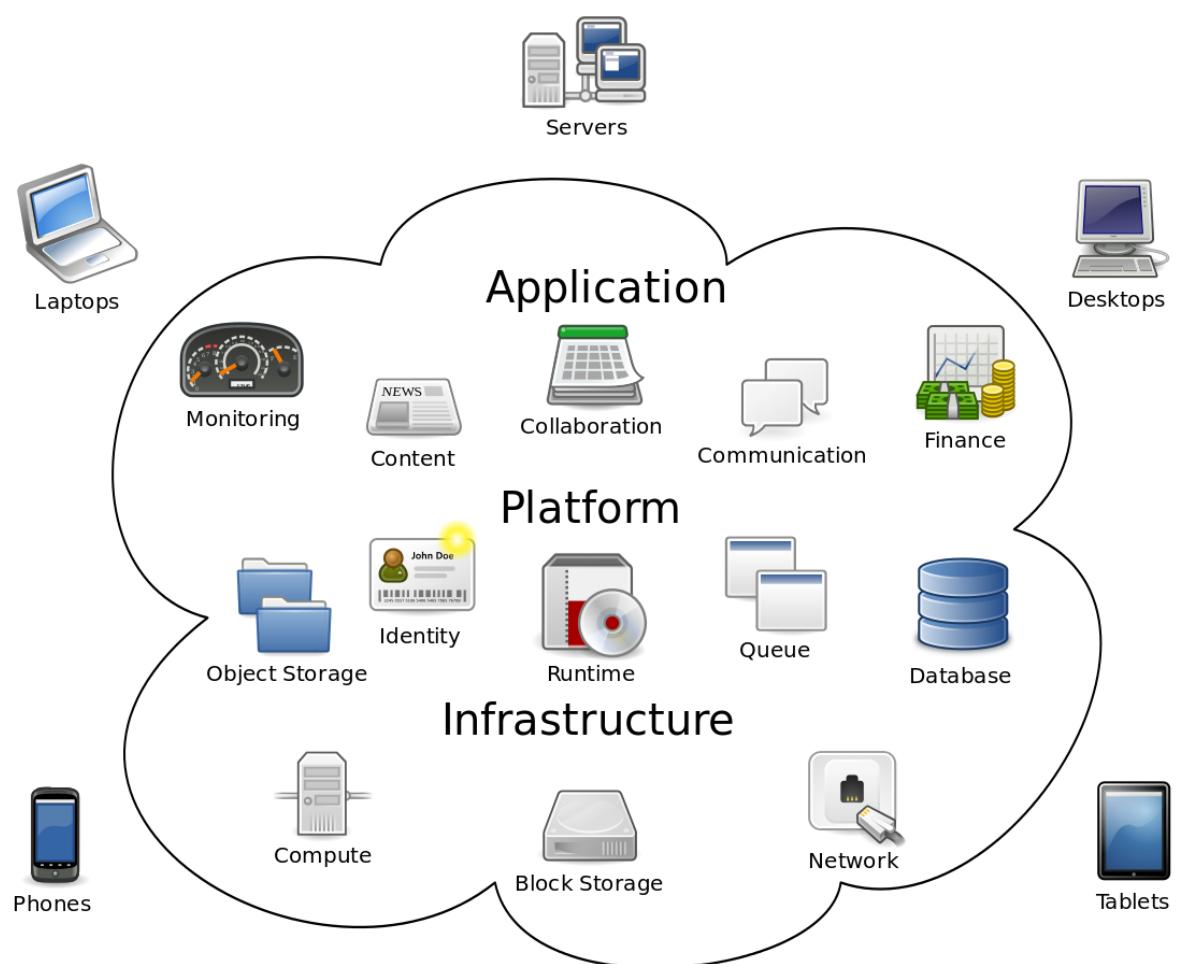
A Übersicht über die Konzernstruktur



B Versionen des SAP ERP



C Überblick über Cloud Computing [74]



D SAP Support-Chat

Info um 11:29, Apr 23:

Sie möchten mehr über unsere Lösungen und Services erfahren? Bitte warten Sie, während wir den nächsten verfügbaren SAP-Mitarbeiter kontaktieren. Alle Angaben, die Sie hier machen, werden ausschließlich für dieses Gespräch genutzt. Näheres entnehmen Sie bitte unserer [Datenschutzerklärung](#).

Sie chatten jetzt mit Herr Krämer.

Angela um 11:29, Apr 23:

Hallo Herr Krämer,

Herr Krämer um 11:29, Apr 23:

Hallo, wie kann ich Ihnen helfen?

Angela um 11:29, Apr 23:

ist es möglich, die SAP Cloud Platform auf einem eigenen Server zu hosten?

Herr Krämer um 11:31, Apr 23:

Grundsätzlich ist dies möglich. Sie benötigen aber die entsprechende Infrastruktur.

Herr Krämer um 11:32, Apr 23:

Hier finden Sie Infos zur Cloud Platfom: <https://cloudplatform.sap.com/index.html>

Angela um 11:33, Apr 23:

was wäre die entsprechende Infrastruktur? Und sehen Sie darin überhaupt einen Vorteil?

Angela um 11:33, Apr 23:

vielleicht reicht es, wenn sie die zweite Frage beantworten :-D

Angela um 11:33, Apr 23:

die erste war etwas zu allgemein.

Herr Krämer um 11:35, Apr 23:

Wenn Sie Ihre Daten nicht der SAP anvertrauen wollen, ist die private Cloud die Möglichkeit. Im Prinzip ist die allgemeine Cloud sinnvoller, da Sie sich nicht um die IT kümmern müssen sondern nur user sind.

Herr Krämer um 11:37, Apr 23:

Wir sprechen bei der Cloud Platform von PaaS, Platform as a Service.

Angela um 11:37, Apr 23:

wie funktioniert das Bezahlmodell, wenn wir die Cloud Platform selbst hosten würden? gibt es ein subscription-based package für diesen Fall?

Herr Krämer um 11:39, Apr 23:

Ich bedauere, Kosten und Lizenzmodelle werden ausschließlich durch unseren Vertrieb kommuniziert.

Angela um 11:39, Apr 23:

schade. trotzdem vielen Dank für Ihre Bemühungen, wie kann ich jemanden erreichen, der mir diese Frage beantworten könnte?

Herr Krämer um 11:40, Apr 23:

Setzen Sie berets SAP Lösungen ein?

Angela um 11:41, Apr 23:

ja, wir sind als IT-Outsourcing Dienstleister auf SAP-Produkte spezialisiert.

Herr Krämer um 11:42, Apr 23:

Sind Sie SAP Partner?

Angela um 11:42, Apr 23:

ja.

Herr Krämer um 11:42, Apr 23:

Dann haben Sie einen Partner Manager. Er wäre der richtige Ansprechpartner.

Angela um 11:43, Apr 23:

Vielen Dank! Ich wünsche Ihnen eine schöne Woche.

Aus <<https://www.sap.com/germany/index.html>>

E Überblick über Testkonten in der SAP Cloud Platform

Trial Accounts

	Cloud Foundry Environment	Neo Environment
Use case	A trial account enables you to explore the basic functionality of the Cloud Foundry environment for 90 days. Access is open to everyone.	A trial account enables you to explore the basic functionality of the Neo environment for a non-committal and unlimited period. Access is open to everyone.
Benefits	<ul style="list-style-type: none"> • Free of charge • Self-service registration • Usage possible for 30 days. You can extend the trial period to a maximum of 90 days, after which your account is automatically deleted. <p>The cockpit shows the time left in a free trial. Once a trial is suspended, you can still log on to it, but you won't be able to use applications or services. Between 30 and 90 days after the creation of the trial, you can renew the trial by using the <i>Extend Free Trial</i> button in the cockpit.</p>	<ul style="list-style-type: none"> • Free of charge • Self-service registration • Usage possible for an unlimited period of time • A trial tenant database on a shared HANA MDC system that you can use for 12 hours. <div style="background-color: #f0e68c; padding: 5px; margin-top: 10px;"> <p>⚠ Restriction</p> <p>After 12 hours, it is shut down automatically to free resources (see Restrictions [page 610]).</p> <p>If you do not use the tenant database for 7 days, it is automatically deleted.</p> </div> <ul style="list-style-type: none"> • Multiple deployed Java applications
Services available	Productive and beta services.	Productive and beta services.

F Rechenzentren als Standort für SAP Cloud Platform [75]



G Definition des Wortes *Principal* in Transaktion SAPTERM

SAP term

Quellsprache: Englisch Zielsprache: Deutsch Suchbegriff: Principal

Suchen

Principal:(5)	principal (BC-SEC)	principal (STM0000004)
principal (SD-FT)	Komponente: BC-SEC	principal (STM0000004)
principal (PY)	Englisch	principal (STM0000004)
principal (BC-SEC)	principal	principal (STM0000004)
principal (FIN-FSCM-TRM-TM)	Security (BC-SEC)	principal (STM0000004)
principal (STM0000004)	Object in SAP User Management Engine, for example, user, user account, group, or role.	principal (STM0000004)
SAPterm	Deutsch	SAPterm
SAPterm	Principal	SAPterm
SAPterm	Sicherheit (BC-SEC)	SAPterm
SAPterm	Objekt der SAP User Management Engine, z.B. ein Benutzer, ein Benutzerkonto, eine Gruppe oder eine Rolle.	SAPterm
SAPterm	System principal	SAPterm
SAPterm	Portal Knowledge Management (EP-KM)	SAPterm
SAPterm	A user who has been assigned system-wide permissions in the KM configuration.	SAPterm
SAPterm	A system principal is defined in the portal user management and may be	SAPterm
SAPterm	<ul style="list-style-type: none">• An individual user• A group• A role	SAPterm
SAPterm	A system principal is typically employed by system services (such as, the index management service) that need certain permissions for all resources.	SAPterm
SAPterm	principal (Abkürzung)	SAPterm
SAPterm	Systembenutzer	SAPterm
SAPterm	Portal Knowledge Management (EP-KM)	SAPterm
SAPterm	Ein Benutzer, dem systemweite Berechtigungen in der KM-Konfiguration zugewiesen wurden.	SAPterm
SAPterm	Systembenutzer werden von Systemservices (z.B. dem Index Management Service) gebraucht, die bestimmte Berechtigungen für alle Ressourcen erfordern. Ein Systembenutzer kann ein individueller Benutzer, eine Gruppe oder eine Rolle sein und muss in der Portalbenutzerverwaltung definiert werden.	SAPterm
SAPterm	Benutzer (Abkürzung)	SAPterm

H Quelltexte zur Implementierung des OData-Dienstes

H.1 Entität Person

Methode	PERSONSET_GET_ENTITYSET	aktiv
<pre>1 method PERSONSET_GET_ENTITYSET. 2 DATA:ls_filter_select_options TYPE /iwbp/s_mgw_select_option, 3 ls_select_option TYPE /iwbp/s_cod_select_option. 4 DATA:ls_selectparams TYPE selopt, 5 lt_selectparams TYPE STANDARD TABLE OF selopt. 6 LOOP AT it_filter_select_options INTO ls_filter_select_options. 7 IF ls_filter_select_options-property EQ 'Partner'. 8 LOOP AT ls_filter_select_options-select_options INTO ls_select_option. 9 ls_selectparams-sign = ls_select_option-sign. 10 ls_selectparams-option = ls_select_option-option. 11 ls_selectparams-low = ls_select_option-low. 12 ls_selectparams-high = ls_select_option-high. 13 APPEND ls_selectparams TO lt_selectparams. 14 ENDLOOP. 15 ENDIF. 16 ENDLOOP. 17 SELECT * FROM ysa_gp INTO CORRESPONDING FIELDS OF TABLE ET_ENTITYSET 18 WHERE partner in lt_selectparams. 19 20 endmethod.</pre>		

Methode	PERSONSET_GET_ENTITY	aktiv
<pre>1 method PERSONSET_GET_ENTITY. 2 data: ls_key like line of it_key_tab. 3 4 read table it_key_tab into ls_key with key name = 'Partner'. 5 if sy-subrc eq 0. 6 select * from ysa_gp into corresponding fields of er_entity 7 where partner = ls_key-value. 8 endselect. 9 else. 10 endif. 11 endmethod.</pre>		

H.2 Entität „Vertragskonto“

Methode	VERTRAGSKONTOSET_GET_ENTITYSET	aktiv
1	<code>method VERTRAGSKONTOSET_GET_ENTITYSET.</code>	
2		
3	<code>DATA:ls_filter_select_options TYPE /iwbp/s_mgw_select_option,</code>	
4	<code> ls_select_option TYPE /iwbp/s_cod_select_option.</code>	
5	<code>DATA:ls_selectparams TYPE selopt,</code>	
6	<code> lt_selectparams TYPE STANDARD TABLE OF selopt.</code>	
7	<code>data: ls_key like line of it_key_tab.</code>	
8		
9	<code>read table it_key_tab into ls_key with key name = 'Partner'.</code>	
10	<code>if sy-subrc eq 0.</code>	
11	<code> select * from ysa_vkgp into corresponding fields of table et_entityset</code>	
12	<code> WHERE partner = ls_key-value.</code>	
13	<code>* If there is no carrier specified, read all the flights</code>	
14	<code>else.</code>	
15	<code> select * from ysa_vkgp into corresponding fields of table et_entityset.</code>	
16	<code>endif.</code>	
17		
18	<code>LOOP AT it_filter_select_options INTO ls_filter_select_options.</code>	
19	<code> IF ls_filter_select_options-property EQ 'Vkont'.</code>	
20	<code> LOOP AT ls_filter_select_options-select_options INTO ls_select_option.</code>	
21	<code> move-corresponding ls_select_option to ls_selectparams.</code>	
22	<code> APPEND ls_selectparams TO lt_selectparams.</code>	
23	<code> ENDLOOP.</code>	
24		
25	<code>ENDIF.</code>	
26		
27	<code>ENDLOOP.</code>	
28		
29	<code>DELETE et_entityset where vkont not IN lt_selectparams.</code>	
30		
31	<code>endmethod.</code>	

Methode	VERTRAGSKONTOSET_GET_ENTITY	aktiv
1	<code>method VERTRAGSKONTOSET_GET_ENTITY.</code>	
2	<code>data: ls_key like line of it_key_tab.</code>	
3		
4	<code>read table it_key_tab into ls_key with key name = 'Vkont'.</code>	
5	<code>if sy-subrc eq 0.</code>	
6	<code> select * from ysa_vkgp into corresponding fields of er_entity</code>	
7	<code> where vkont = ls_key-value.</code>	
8	<code> endselect.</code>	
9	<code>else.</code>	
10		
11	<code>endif.</code>	
12	<code>endmethod.</code>	

I Informationen zur Version des SAPUI5-Frameworks im SGW-System

SAP UI development toolkit for HTML5 from ABAP server

This Application provides a ready-to-consume version of SAPUI5.
The SAPUI5 sap-ui-core.js bootstrap file is available at: /sap/public/bc/ui5_ui5/1/resources/sap-ui-core.js

The following libraries are available:

Library	Version: none	Version: 1.22 (1.22.14)	Version: 1.24 (1.24.8)	Version: 1.26 (1.26.18)	Version: 1.28 (1.28.39)
com.sap.apf.apf-lib		1.22.2	1.24.3	1.26.7	1.28.16
com.sap.ca.scfld.md		1.22.4	1.24.7	1.26.5	1.28.13
com.sap.ca.ui		1.22.4	1.24.7	1.26.5	1.28.13
com.sap.collaboration.collaboration-comm		1.22.3	1.24.3	1.26.5	1.28.10
com.sap.ip.bi.zen.rt.components.crosstab	15.1.0.201509240501				
com.sap.ip.bi.zen.rt.dsh.phx	15.1.0.201605012045				
com.sap.ip.bi.zen.rt.themelib.phx	15.1.0.201509240501				
com.sap.lm.uiext.landvisz.sap.landvisz			1.24.0	1.26.1	1.28.0
com.sap.portal.ui5.portal-ui5-ext		1.22.2	1.24.0	1.26.0	1.28.0
com.sap.smartbusiness.suite.smartbusines		1.22.3	1.24.3	1.26.5	1.28.26
com.sap(suite).suite-ui-commons		1.22.3	1.24.3	1.26.6	1.28.25
com.sap.ui5.common		1.22.13	1.24.7	1.26.16	1.28.38
com.sap.ui5.comp		1.22.13	1.24.7	1.26.16	1.28.38
com.sap.ui5.core		1.22.13	1.24.7	1.26.16	1.28.38
com.sap.ui5.dist.fiori-lib		1.22.14	1.24.8	1.26.18	1.28.39
com.sap.ui5.fl				1.26.16	1.28.38
com.sap.ui5.inbox		1.22.13	1.24.7	1.26.16	1.28.38
com.sap.ui5.layout		1.22.13	1.24.7	1.26.16	1.28.38
com.sap.ui5.makit		1.22.13	1.24.7	1.26.16	1.28.38
com.sap.ui5.mobile		1.22.13	1.24.7	1.26.16	1.28.38
com.sap.ui5.mobile-ext		1.22.13	1.24.7	1.26.16	1.28.38
com.sap.ui5.ndc					1.28.38
com.sap.ui5.richtexteditor		1.22.13	1.24.7	1.26.16	1.28.38
com.sap.ui5.suite		1.22.13	1.24.7	1.26.16	1.28.38
com.sap.ui5.table		1.22.13	1.24.7	1.26.16	1.28.38
com.sap.ui5.themelib_sap_bluecrystal		1.22.13	1.24.7	1.26.16	1.28.38
com.sap.ui5.themelib_sap_goldreflection		1.22.13	1.24.7	1.26.16	1.28.38
com.sap.ui5.themelib_sap_platinum		1.22.13	1.24.7	1.26.16	1.28.38
com.sap.ui5.unified		1.22.13	1.24.7	1.26.16	1.28.38
com.sap.ui5.ux3		1.22.13	1.24.7	1.26.16	1.28.38
com.sap.ui5.vbm		1.22.6	1.24.2	1.26.6	1.28.8
com.sap.ui5.viz		1.22.13	1.24.7	1.26.16	1.28.38
com.sap.ushell.ushell-lib		1.22.9	1.24.7	1.26.10	1.28.35
com.sap.ushell.ushell_abap	1.28.34				
com.sap.uxap.uxap-uilib				1.26.6	1.28.10
sap.epm.ui.controls	1.22.2				
sap.epm.ui.controls.mobile	1.16.2				

For more information and documentation look at the SAPUI5 Demokit

[Demokit](#)

[Demokit](#)

[Demokit](#)

[Demokit](#)

You can use SAPUI5 in any HTML file with the following script tag:

```
<script id="sap-ui-bootstrap"
type="text/javascript"
src="/sap/public/bc/ui5_ui5/1/resources/sap-ui-cachebuster/sap-ui-core.js"
data-sap-ui-theme="sap_goldreflection"
data-sap-ui-libs="sap.ui.commons">
</script>
```

J Anpassen der Anwendungsvorlage

J.3 SAP Web IDE in SAP Cloud Platform – CRUD Master-Detail Application

New CRUD Master-Detail Application

Template Customization

Data Binding - Object	Object Collection*	PersonSet
	Object Collection ID*	Partner
	Object Title*	Partner
	Object Numeric Attribute	
	Object Unit of Measure	
<input checked="" type="checkbox"/> Display only required fields when creating a new entity.		
Data Binding - Line Item	Line Item Collection	VertragskontoSet
	Line Item Collection ID	Vkont
	Line Item Title	Vkbez
	Line Item Numeric Attribute	
	Line Item Unit of Measure	

[Previous](#)

[Next](#)

[Finish](#)

J.4 SAP Web IDE Personal Edition – Master Master-Detail Application

New SAP Fiori Master Master-Detail Application

Template Customization

Master 1 Section	Title	Partner anzeigen
	OData Collection	PersonSet
	Search Placeholder	Geschäftspartner
	Search Tooltip	Bitte geben Sie die Nummer eines Geschäftspartners ein.
	Search Field	Partner
Master 1 Data Fields	Item Title	Partner
	Attribute 1	Attribute from the OData collection
	Attribute 2	Attribute from the OData collection
Master 2 Section	Title	Vertragskonten
	OData Collection Navigation	VertragskontoSet
	Search Placeholder	Vertragskontonummer
	Search Tooltip	Bitte geben Sie die Nummer eines Vertragskontos ein.
	Search Field	Vkont
Master 2 Data Fields	Item Title	Vkont
	Numeric Attribute	Numeric attribute from the target collection
	Unit of Measure	Unit of Measure from the target collection
	Status Attribute	Status attribute from the target collection
	Attribute	Additional attribute from the target collection
Detail Section	Title	Informationen zum Vertragskonto
	Attribute 1	Vkbez
	Attribute 2	Vkont
	Attribute 3	Partner

[Previous](#)[Next](#)[Finish](#)

K Ansicht der Anwendung nach der Erstellung

K.5 SAP Web IDE in SAP Cloud Platform – CRUD Master-Detail Application

The screenshot shows a SAP Web IDE interface for a CRUD Master-Detail Application. The left sidebar lists four entries under 'PersonSet (4)': 1000010000, 1000020000, 1000030000, and 1000040000. The main panel displays the details for the first entry, '1000010000'. It includes a search bar, a back arrow, a SAP logo, and a 'Partner anzeigen' dropdown. Below the entry ID are two circular icons: one with an 'i' for information and one with a clipboard for copy. A section titled 'VertragskontoSet (3)' lists three entries: 'Lausner Weg 43' (with ID 100100100100), 'Lausner Weg 44' (with ID 100100100200), and 'Lausner Weg 45' (with ID 100100100300). At the bottom right are 'Edit', 'Delete', and a refresh icon.

K.6 SAP Web IDE Personal Edition – Master Master-Detail Application

Partner anzeigen

Geschäftspartner

1000010000	
1000020000	
1000030000	
1000040000	

← Vertragskonten

Vertragskontonummer

100100100100	
100100100200	
100100100300	

Informationen zum Vertragskonto

100100100100

Vkbez:
Lausner Weg 43

Vkont:
100100100100

Partner:
1000010000

L Verteilen einer Anwendung in der SAP Cloud Platform

Deploy Application to SAP Cloud Platform X

A build will be triggered for the project. The build results will be deployed.

Application Details

Deploy a new application Update an existing application

*Account: s0019333218trial Get Accounts

Project Name: bp_crud

*Application Name: bpcrud

Version Management

*Version: 1.0.0 Activate

Deploy Cancel

Successfully Deployed X

Version 1.0.0 has been created.

[Open the active version of the application](#)

[Open the application's page in the SAP Cloud Platform cockpit](#)

You can now register the application to SAP Fiori launchpad.

Register to SAP Fiori launchpad Close

M Anlegen eines Fiori-Launchpad im Portal-Dienst der SAP Cloud Platform

M.7 Anlage des Katalogs

The screenshot shows the Fiori Configuration Cockpit interface for 'Angela's Launchpad'. On the left, a sidebar lists navigation options: Startseite, Einstellungen, Inhaltsverwaltung (selected), Apps, Kataloge, Gruppen, Rollen, Services und Tools, App-Ressourcen, Verwendungsanalyse, Nützliche Tools, Nützliche Links, and Rechtliche Informationen. The main area displays a list of catalogs under 'Kataloge (3)'. The first catalog, 'Angelas Katalog' (Catalog for Bachelorarbeit), is selected. It shows three tabs: 'Eigenschaften' (selected), 'Apps', and 'Rollen'. The 'Allgemein' section contains details: Catalogname: Angela's Katalog, Catalog-ID: aa2193da-088d-4c6b-8acd-810e9d220fec-1528902807029, and Beschreibung: Catalog for Bachelorarbeit. The 'Zusatzinformationen' section shows creation and last update details. At the bottom right are 'Bearbeiten' and 'Löschen' buttons.

This screenshot shows the properties page for 'Angelas Katalog'. It features three tabs: 'Eigenschaften' (selected), 'Apps', and 'Rollen'. The 'Rollen' tab is highlighted with a dashed blue border. Below the tabs, the title 'Angelas Katalog' is displayed.

Zugeordnete Rollen (1)

Rollenname	Rollen-ID
Portal_User	Portal_User

M.8 Anlage einer Gruppe

Angelas Gruppe



Allgemein

Gruppenname: Angelas Gruppe

Gruppen-ID: 4e70ebd8-b221-40c6-8226-363628ddf37f-1528903015105

Gesperre Gruppe

Zusatzinformationen

Angelegt: 17:19 13.6.2018

Angelegt von: s0019333218

Zuletzt geändert: 17:21 13.6.2018

Zuletzt geändert von: s0019333218

Angelas Gruppe



Zugeordnete Rollen (1)

Rollenname	Rollen-ID
Portal_User	Portal_User

N Registrierung einer Anwendung auf einem Launchpad in der SAP Cloud Platform

General Information > Tile Configuration > Assignment > Confirmation

Register to SAP Fiori Launchpad

General Information

*Provider Account: trial (flpportal)

*Application Name: angelabp_crud

Description: Geschäftspartner anzeigen

Intent:

<input checked="" type="checkbox"/>	Semantic Object	Action
<input checked="" type="checkbox"/>	bp_crud	Display

More information on intent properties.

Previous Next

General Information > Tile Configuration > Assignment > Confirmation

Register to SAP Fiori Launchpad

Tile Configuration

*Type: Dynamic

*Title: Geschäftspartner anzeigen

Subtitle:

Icon: sap-icon://person-placeholder

Browse

*Collection: PersonSet

Preview:

Geschäftspartner anzeigen

4

Geschäftspartner

Previous Next

General Information Tile Configuration **Assignment** Confirmation

Register to SAP Fiori Launchpad

Assignment

*Site

*Catalog

*Group

[Previous](#) [Next](#) [Finish](#)

SAP Startseite ▾

Angelas Gruppe

Geschäftspartner anzeigen

4 Geschäftspartner

O Verteilen einer Anwendung auf einem lokalen Applikationsserver

Deployment Options Deploy a New Application Select a Transport Request Confirmation

Deploy to SAPUI5 ABAP Repository

Deployment Options

System * SGW-destination

Deploy a new application
 Update an existing application

i The SAPUI5 versions of your application and the selected SAP system are compatible.
Application:1.28.40
SAP system:1.28.38

Previous Next

Deployment Options Deploy a New Application Select a Transport Request Confirmation

Deploy to SAPUI5 ABAP Repository

Deploy a New Application

Name *

Description *

Package *

Previous Next

Deployment Options Deploy a New Application **Select a Transport Request** Confirmation

Deploy to SAPUI5 ABAP Repository

Select a Transport Request

Choose from requests in which I am involved

Request Number
SGWK900181

Transport Request	User	Target	Description
SGWK900181	STOECKAN	/ZSGW/	Auftrag Bachelorarbeit Angela

Create a new request

Request Description
The Request ID is automatically generated

Enter a request number

Request Number

[Previous](#) [Next](#)

P Registrieren einer Anwendung auf einem lokalen Fiori-Launchpad

P.9 Ansicht Anwendungsspeicher auf Applikationsserver

Paket	
YSA_ABAP_SAPUI5_REPOSITORY	
Objektname	Beschreibung
YSA_ABAP_SAPUI5_REPOSITORY	eigenes Paket als ABAP repository
BSP-Bibliothek	
BSP-Applikationen	
YSA_BP	Geschäftspartner anzeigen Angela
YSA_BP_MMD	Geschäftspartner anzeigen

P.10 Semantisches Objekt anlegen

Sicht "Semantische Objekte - Kundenansicht" anzeigen: Übersicht

Semantische Objekte - Kundenansicht		
Semantisches Objekt	Name semantisches Objekt	Beschreibung semantisches Objekt
SEMOBJ_YSA_BP	SEMOBJ_YSA_BP	Semantic Object for YSA_BP
SEMOBJ_YSA_BP_MMD	SEMOBJ_YSA_BP_MMD	Semantic Object for YSA_BP_MMD

P.11 Benutzerrolle mit Zuordnung von Katalog, Gruppe und Nutzer

The screenshot shows the configuration of a user role:

- Role:** YSAROLE (checkbox 'Veraltet' is unchecked)
- Kurzbeschreibung:** Angelas Rolle
- Zielsystem:** Keine Destination
- Hierarchie:** Menü der Rolle (with entries: Angelas Katalog, Angelas Gruppe)
- Knotendetails:**
 - Typ: SAP Fiori Kachelkatalog
 - Objekt: X-SAP-UI2-CATALOGPAGE:YSA_KATALOG?AUTH_DEFAULTS=
 - Text: Angelas Katalog

Benutzerzuordnungen				
Benutzerkennung	Benutzername	von	bis	I...
STOECKAN	Angela Stöckert	23.05.2018	31.12.9999	

P.12 Konfiguration des Launchpad im Applikationsserver

Launchpad ändern - Rolle: YSALPD, Instanz: TRANSACTIONAL (DE)

Link-Informationen
Link-Text: Geschäftspartner anzeigen
Beschreibung:

Anwendungstyp
Anwendungstyp: URL

Anwendungsparameter
URL: /sap/bc/ui5_ui5/sap/ysa_bp_mmd

System
Systemalias:

Erweiterte (optionale) Parameter ausblenden

Erweiterte Parameter (optional)
Anwendung - Deaktivierung durch Benutzer
Anwendung kann nicht aus LPD entfernt werden:

Anwendungsspezifische Parameter
Anwendungsalias: geschaeftpartner_anzeigen

geschaeftpartner_anzeigen

Ze 1, Sp 1 Ze 1 - Ze 1 von 1 Zeilen
Zielanwendungsparam.
Aussetzen/fortsetzen
Proxy-Klasse
Zusatzinformationen: SAPUI5.Component=angela
FPM-Ereignis-ID

Portalparameter
Navigationsmodus: EXT_HEAD Portal-Fenster ohne Ko...
Fenstereigenschaften
Historiemodus: 1 Navigationseintrag kann in Histori...
Parameterweiterleitung: G Parameter holen

P.13 Kachelkatalog mit Zielverknüpfungen im Launchpad Designer

The screenshot shows the SAP Launchpad Designer interface. On the left, there's a sidebar with 'Catalogs' and 'Groups' tabs, and a search bar for catalogs. Below it is a list of catalog entries:

- Angelas Katalog (YSA_KATALOG) - 6 items
- Außendienstmitarbeiter (SD) - 11 items (SAP_SD_BC_FIELDSALESREP_X1)
- Buyer (MM) - Content - 4 items

The main area displays a catalog entry for 'Angelas Katalog' with ID X-SAP-UI2-CATALOGPAGE:YSA_KATALOG. It shows three tiles: 'Tiles' (4 items), 'Tiles' (4 items), and 'Target Mapp...' (2 items). Below the tiles is a table with columns: Icon, Title, Semantic Object, Action, Parameters, Target URL, Reference, and Outd... . The table lists four entries, all related to displaying business partners.

Icon	Title	Semantic Object	Action	Parameters	Target URL	Reference	Outd...
	Geschäftspartner anzeigen	SEMOBJ_YSA_BP_MMD	display		#SEMOBJ_YSA_BP_MMD-display		
	Geschäftspartner anzeigen	SEMOBJ_YSA_BP	display		#SEMOBJ_YSA_BP-display		
	Geschäftspartner anzeigen	SEMOBJ_YSA_BP	display		#SEMOBJ_YSA_BP-display		
	Geschäftspartner anzeigen	SEMOBJ_YSA_BP_MMD	display		#SEMOBJ_YSA_BP_MMD-display		

At the bottom, there are buttons for Create, Configure, Create Reference, Delete, and Where-Used.

P.14 Gruppe mit Anwendungskacheln im Launchpad Designer

The screenshot shows the SAP Launchpad Designer interface. On the left, there's a sidebar with 'Catalogs' and 'Groups' tabs, and a search bar for groups. Below it is a list of group entries:

- Angelas Gruppe (YSA_GROUP) - 4 items
- arvato Systems Applikationen für Z_GROUP_ISU_APPS - 3 items
- arvato Systems Vertriebsmitarbeiter TEST_VERTREIB_01 - 5 items
- Buyer (MM) (SAP_MM_BCG_BUYER) - 3 items
- Buyer (MM) (SAP_MM_BCG_Buyer_X1) - 5 items
- Cost Manager (SAP_FIN_BCG_CostManager_X1) - 1 item

The main area displays a group entry for 'Angelas Gruppe' with ID YSA_GROUP. It shows four application tiles arranged in a 2x2 grid:

- Geschäftspartner anzeigen (blue user icon)
- Geschäftspartner anzeigen (blue user icon)
- Geschäftspartner anzeigen alte Version (blue user icon)
- Geschäftspartner anzeigen alte Version (blue user icon)

Each tile has a large numerical value (1,23K) and the text 'Geschäftspartner' below it.

12 Selbstständigkeitserklärung

Ich versichere, dass ich die vorliegende Arbeit ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht. Die Arbeit wurde bisher in gleicher oder ähnlicher Form weder veröffentlicht, noch einer anderen Prüfungsbehörde vorgelegt.

Leipzig, 26.06.2018

Ort, Datum

Unterschrift