

# Einleitung

Die Anwendungsentwicklung im SAP-Umfeld war lange geprägt vom klassischen On-Premise-Deployment-Paradigma der Anwendungssoftware. Hier entwickeln Kunden und Partner eigenständige Erweiterungen der SAP-Standardprodukte, meist auf Basis des ABAP-Applikationsservers. Dieser stellt Entwicklungsumgebung, Modularisierungs- und Schnittstellentechnologie sowie Software-Lifecycle-Management-Werkzeuge zur Verfügung, um eine Erweiterung optimal zu integrieren.

Mit der wachsenden Popularität von Cloud-Plattformen und dem Software-as-a-Service-Vertriebs- und -Benutzungsmodell kann sich SAP nicht mehr alleine auf ein geschlossenes, proprietäres Programmiermodell und Erweiterungsmodell verlassen. Kunden und Partner wollen neue, sich agil entwickelnde Technologien im Cloud-Umfeld nutzen, wo sie aber auf völlig andere Konzepte und Architekturen treffen als in der klassischen On-Premise-Welt. *Microservices, Cloud Native Design, 12 Factor Applications, Continuous Deployment* und *DevOps* sind nur einige Konzepte, die sich im Umfeld der Cloud-Plattformen entwickelt haben, oder auch nur eine Renaissance unter neuem Namen erleben. Viele frei verfügbare Frameworks und Werkzeuge für die verschiedenen Programmiersprachen konkurrieren um die Gunst der Entwickler\*innen, besitzen allerdings oft nur eine Lebensdauer von wenigen Jahren.

Mit dem Zugang zur SAP Cloud Platform stehen SAP-Kunden und Partner deshalb vor folgenden Fragen:

- Wie entwickeln wir cloud-native Anwendungen, die unsere bestehenden SAP-Anwendungen und -Prozesse erweitern?
- Wie entwickeln wir neue, differenzierende Geschäftsanwendungen, die sich optimal in unseren bestehenden SAP-Kontext integrieren?
- Wie profitieren wir von der rasanten Technologieentwicklung im Cloud-Umfeld, ohne dass sich unser Entwicklungsteam mehr mit technologischen Neuerungen auseinandersetzt als mit der Implementierung eines geschäftlich relevanten Prozesses?
- Wie befreien wir unsere Entwicklungsabteilung von der Last der immer wiederkehrenden Implementierung grundlegender Qualitäten wie Sicherheit, Skalierbarkeit, Erfüllung von Datenschutzvorgaben, Auditierbarkeit, Internationalisierung etc.?

Eine Antwort von SAP auf diese Fragen ist das *SAP Cloud Application Programming Model (CAP)*. CAP wird auch als Backend-Programmiermodell be-

Herausforderungen  
der Software-  
entwicklung

zeichnet. Dies bedeutet, dass es kein Modell zur Entwicklung von Benutzeroberflächen darstellt. Über die Benutzeroberfläche macht CAP im Kern keine Annahmen. SAP Fiori kann aber besonders einfach über das OData-Protokoll als UI-Programmiermodell zusammen mit CAP eingesetzt werden.

**Ziele des Buches** Ziel dieses Buches ist es, dass Sie anhand von Implementierungsbeispielen die Grundprinzipien von CAP verstehen und auch seine Abhängigkeiten zur Welt der SAP Cloud Platform. Ich möchte Sie in die Lage versetzen, mittels der schnell wachsenden CAP-Community und der von SAP zur Verfügung gestellten Ressourcen Ihre eigenen CAP-Projekte erfolgreich durchführen zu können.

Sie werden bemerken, dass ich in diesem Buch großen Wert auf den Praxisbezug lege. Anhand einer Beispielanwendung werden Sie Schritt für Schritt in immer fortgeschrittenere Aspekte von CAP eingeführt. Diese Teile werden durch eher theoretische Abschnitte ergänzt, sodass Sie auch eine Vorstellung von den Möglichkeiten bekommen, die naturgemäß im Rahmen einer Beispielanwendung nicht demonstriert werden können.

Es ist nicht das Ziel dieses Buches, eine möglichst vollständige Faktensammlung aller Details zu CAP zu sein. Dieser Versuch müsste scheitern, da ständig neue Fähigkeiten und Integrationspunkte entstehen. Vielmehr sollen Sie anhand eines nachvollziehbaren Beispiels die stabilen Grundprinzipien von CAP verstehen und die sich darum gruppierenden Werkzeuge.

**Zielgruppe** Dieses Buch richtet sich vor allem an Architekt\*innen und Entwickler\*innen, die Anwendungen auf der SAP Cloud Platform entwerfen und implementieren, aber auch an für Entwicklungsteams verantwortliche Manager\*innen, die nach Ansätzen zur Applikationsentwicklung für ihre Teams suchen. Ich setze deswegen grundlegende Kenntnisse der Softwareentwicklung voraus. Obwohl nicht zwingend notwendig, ist es von Vorteil, wenn Sie insbesondere Vorkenntnisse im Bereich moderner Entwicklungswerkzeuge (z. B. Eclipse, Visual Studio Code), im Umgang mit Command-Line-Werkzeugen (Microsoft PowerShell oder Linux Bash), der relationalen Datenmodellierung, der Cloud Foundry Platform-as-a-Service und der Programmierung in Node.js mitbringen. Meine Entscheidung, Node.js zu favorisieren, liegt schlicht darin begründet, dass die Node.js-Ausprägung von CAP früher produktiv einsetzbar war als die Java-Variante und auch in der SAP-Community verbreiteter ist.

Falls Sie nicht in allen Bereichen Vorkenntnisse haben, seien Sie unbesorgt. Das Buch führt Sie im Stil eines Tutorials durch die Entwicklung einer Beispielanwendung mittels CAP, ohne technisch zu detailliert zu werden. Da-

mit sollen insbesondere auch Entwickler\*innen angesprochen werden, die aus der klassischen SAP-Umgebung des ABAP-Applikationsservers kommen. CAP versucht nicht, diese Umgebung zu kopieren. Dies würde aufgrund der verschiedenen technischen Randbedingungen und Paradigmen gar keinen Sinn ergeben. Trotzdem werden Sie sich in den Prinzipien von CAP wiederfinden, die es erlauben, Ihre Kompetenz als in der Entwicklung betriebswirtschaftlicher Anwendungen voll zu nutzen.

CAP ist ein *Backend-Programmiermodell*, d. h., es adressiert vor allem, aber nicht notwendigerweise, die Entwicklung von Services mit angeschlossener Datenbank. CAP als Programmiermodell für die SAP Cloud Platform integriert sich natürlich mit anderen SAP-Technologien. Vor allem sind dies die Cloud-Foundry-Umgebung der SAP Cloud Platform, SAP Fiori und SAP HANA. All dies sind Bereiche, die eigene Prinzipien und Programmiermodelle exponieren. Dieses Buch wird jeweils Hintergrundinformationen zu diesen Technologien liefern, soweit sie für die Integration mit CAP wichtig sind und die gezeigten Beispiele es erfordern. Das Vermitteln solcher Technologien im vollem Umfang sprengt allerdings den Rahmen des Buches. Der Rheinwerk Verlag bietet Ihnen jedoch auch zu diesen Themen Vertiefungsmöglichkeiten.

Um die hier entwickelten Beispiele nachvollziehen zu können, sollten Sie über einen Computer mit Zugang zum Internet verfügen. Auf diesem Computer benötigen Sie die Berechtigung, Software installieren zu können. Sie werden nur gängige, freie Software benötigen, für die keine Lizenzen erforderlich sind. Um auch die letzten Kapitel des Buches nachvollziehen zu können, benötigen Sie allerdings einen Zugang zur SAP Cloud Platform. Falls dieser nicht schon im Kontext Ihres Unternehmens existiert, können Sie über einen Trial-Account Zugang erhalten.

## Aufbau des Buches

**Teil I** besteht aus zwei einführenden Kapiteln. **Kapitel 1**, »Einführung in das SAP Cloud Application Programming Model«, ist theoretischer Natur und beantwortet vor allem die Fragen, was das SAP Cloud Application Programming Model ist und was seine Bestandteile und Prinzipien sind. Da CAP genutzt wird, um Anwendungen für die SAP Cloud Platform zu bauen, werden kurz deren Zielsetzung und Eigenschaften rekapituliert. Die SAP Cloud Platform kann im Rahmen dieses Buches nicht im Detail behandelt werden. Trotzdem ist es wichtig, die Beziehung von CAP zur SAP Cloud Platform genau zu verstehen, ebenso wie die Beziehungen zu SAP Fiori und SAP HANA.

Was Sie hier nicht finden werden

Technische Voraussetzungen

Teil I: Motivation und Einführung

Mit **Kapitel 2**, »Erste Schritte zur eigenen Anwendung«, wechseln Sie in den Praxismodus. Zuallererst beschreibe ich, welche Software Sie installieren müssen und wo Sie diese aus dem Internet beziehen können. Anhand eines "Hello World"-Beispiels werden einige CAP-Komponenten beschrieben und angewandt. Mithilfe dieses kleinen Beispiels werden Sie in die Art und Weise eingeführt, wie Sie mit CAP Entwicklungsprojekte durchführen sollten.

**Teil II: Entwicklung einer Beispielanwendung**

**Teil II** beschäftigt sich mit der Entwicklung einer relevanteren Anwendung. Sie werden zur Erstellung eines Produktkatalogs ein typisches CAP-basiertes Entwicklungsprojekt aufsetzen und strukturieren. Die Beispielanwendung wird im gesamten Teil II dieses Buches rein lokal auf Ihrem Rechner entwickelt. Sie brauchen dafür noch keinen Zugang zur SAP Cloud Platform.

In **Kapitel 3**, »Start des Entwicklungsprojekts«, wenden Sie die in Teil I erlernten Techniken auf die Entwicklung einer realistischen Anwendung, einen Produktkatalog, an. Dieses Kapitel beschreibt die Erstellung der ersten Version dieser Anwendung. Darüber hinaus zeige ich Ihnen das Testen mittels HTTP-Clients sowie das Debuggen Ihrer Anwendung.

**Kapitel 4**, »Core Data Services für CAP im Detail«, beschäftigt sich systematisch mit der Modellierungssprache CDS für CAP. Sie lernen die Grundprinzipien, aber auch weiterführende Konzepte von CDS kennen. Betrachten Sie dieses Kapitel als theoretische Vertiefung. Nicht alle Techniken von CDS werden Sie für die Beispielanwendung benötigen. Für die zukünftige Implementierung Ihrer eigenen Anwendungen finden Sie hier aber eine kompakte Beschreibung der Sprache CDS.

Mithilfe von **Kapitel 5**, »Weiterentwicklung zur produktiven Anwendung«, werden Sie die Beispielanwendung sukzessive mit fortgeschrittenen CDS-Techniken ausstatten, die Sie für einen produktiven Einsatz im Geschäftsumfeld benötigen. Es werden Eigenschaften wie Internationalisierung, Authentifizierung, Autorisierung und die Wiederverwendung von Modellen behandelt. Die Verwendung von SAP-Fiori-Benutzeroberflächen im Zusammenhang mit CAP innerhalb Ihrer lokalen Testumgebung wird im Detail erläutert.

In **Kapitel 6**, »Das CDS API für Node.js«, werden Sie sich erneut mit Theorie beschäftigen dürfen. Sie lernen die Schnittstelle kennen, mit der Sie programmatisch in Node.js mit CDS interagieren können. Allerdings ist dieses Kapitel auch für Java-Programmierer\*innen relevant, da nicht nur technische Schnittstellen, sondern auch die Einsatzmöglichkeiten und die semantischen Eigenschaften der Schnittstelle besprochen werden. Auch bezüglich

der API-Methoden ist die Java-Schnittstelle mit der Node.js-Schnittstelle vergleichbar.

In **Teil III** dreht sich alles um das Entwickeln und Deployen einer CAP-basierten Anwendung im Kontext der SAP Cloud Platform. Bis zu diesem Punkt werden Sie Plattform-Services wie die SAP-HANA-Datenbank oder den Service zur Authentifizierung nur lokal simuliert haben. Diese werden nun durch entsprechende produktive Services der SAP Cloud Platform ersetzt. In einem Zwischenschritt kann der Applikationsserver weiterhin in Ihrer lokalen Umgebung verbleiben. Als Endausbaustufe wird Ihre Anwendung komplett auf die SAP Cloud Platform deployt sein.

**Kapitel 7**, »Deployment-Umgebung auf der SAP Cloud Platform anlegen«, zeigt, wie Sie Zugang zu einem Trial-Account der SAP Cloud Platform bekommen und welche Konfigurationen Sie vornehmen müssen, um die Cloud-Foundry-Umgebung einzurichten. Im Zuge einer Einführung in die Arbeit mit Cloud Foundry deployen Sie eine Anwendung in die Cloud-Foundry-Umgebung.

In **Kapitel 8**, »SAP-Anwendungsarchitektur in der Cloud-Foundry-Umgebung«, steht die Blaupause von SAP für Geschäftsanwendungen in der Cloud-Foundry-Umgebung der SAP Cloud Platform im Mittelpunkt. Sie lernen die Application-Router-Komponente kennen, die unter anderem als Frontend-Server für die Benutzeroberfläche fungiert. Die Rolle des SAP HANA Cloud Services sowie des SAP Cloud Platform Authorization and Trust Management wird erläutert. Ihre Beispielanwendung wächst dabei um eine SAP-Fiori-Benutzerschnittstelle.

**Kapitel 9**, »Anbindung an SAP HANA«, beschäftigt sich speziell mit der Beziehung von CAP zum SAP HANA Cloud Service, sowohl während der lokalen Entwicklung als auch nach einem produktiven Deployment.

Das Szenario der hybriden Entwicklung wird in **Kapitel 10**, »Hybride Entwicklung«, behandelt. Die Applikationsserver einer Anwendung können dabei in der lokalen Entwicklungsumgebung verbleiben, kommunizieren aber dennoch mit Services auf der SAP Cloud Platform. Dies reduziert die Dauer von Entwicklungs- und Testzyklen erheblich. Neben SAP HANA steht die Konfiguration und Nutzung des SAP Cloud Platform Authorization and Trust Management Services im Mittelpunkt.

Das eigentliche Deployment und der Betrieb der Anwendung auf der SAP Cloud Platform wird Ihnen in **Kapitel 11**, »Deployment der Geschäftsanwendung«, vorgestellt. Zwei Alternativen werden präsentiert und verglichen: zum einen die Nutzung des Cloud-Foundry-Kommandozeilenwerkzeugs,

**Teil III: Arbeiten mit der SAP Cloud Platform**

zum anderen die Technik der Multitarget Applications. Zum Ende dieses Kapitels ist Ihre Anwendung vollständig auf der SAP Cloud Platform verfügbar.

Kaum ein Cloud-Service arbeitet isoliert von anderen Produkten und Services. **Teil IV** zeigt Ihnen daher, wie Sie Services der SAP Cloud Platform nutzen können, um CAP-basierte Anwendungen zu entwickeln, aber vor allem zur Laufzeit mit Produkten des SAP-Portfolios zu integrieren.

**Kapitel 12**, »Das SAP Business Application Studio«, stellt Ihnen *IDE as a Service* auf der SAP Cloud Platform vor, was unter anderen für die Entwicklung von CAP-basierten Anwendungen genutzt werden kann.

In **Kapitel 13**, »Integration mit SAP-Lösungen«, erhalten Sie einen Überblick über die Kommunikation von CAP-basierten Anwendungen mit anderen Lösungen von SAP. Sie lernen Techniken und Services kennen, mit denen Sie Ihre CAP-basierte Anwendung beispielsweise mit SAP S/4HANA verbinden können. Für eine ereignisbasierte Kommunikation kommt dafür das SAP Cloud Platform Enterprise Messaging zum Einsatz.

**Kapitel 14**, »Problembehandlung«, bietet Ihnen Hilfestellungen, wenn es mal mit dem einen oder anderen Schritt des Entwicklungsprozesses nicht gleich klappt.

Zum Schluss gibt Ihnen **Kapitel 15**, »Zusammenfassung und Ausblick«, ein Resümee und einen Ausblick auf kommende Neuerungen im Umfeld der Anwendungsentwicklung auf der SAP Cloud Platform.

## Danksagung

Auch wenn dieses Buch nur durch einen einzelnen Autor erstellt wurde, so ist es doch ein Kondensat von Ideen und ihren Umsetzungen, die mit der Zeit von vielen beigetragen wurden. Allen konstruktiv Beitragenden gilt mein Dank. Neben dem gesamten CAP-Team möchte ich aber dennoch die, aus meiner Sicht, drei »Macher« des SAP Cloud Application Programming Model in der heutigen Form besonders hervorheben, mit denen ich glücklicherweise eine Zeitlang eng zusammenarbeiten durfte: Zualererst danke ich dem »Erzeuger« Daniel für seine Sturheit, Konsequenz und Brillanz, ohne die sich CAP schon lange in ein verwaschenes Konzept aufgelöst hätte. Dem »Geburtshelfer« Johannes gilt es, für seine Weitsicht und Geduld mit Einzelnen zu danken. Ohne ihn hätte CAP niemals organisatorisch Fahrt aufgenommen. Und dann ist da noch der »Taufpate« und »Missionar«,

Christian, der CAP beim Aufwachsen zur Seite steht, wenn es gilt, fachliche und organisatorische Probleme aus dem Weg zu räumen.

Danken möchte ich auch dem Lektorat des Rheinwerk Verlags, das mich so effektiv und unkompliziert bei meinem ersten Buch dieser Art geleitet hat.

**Dr. Klaus Kopecz**

September 2020