

Diese Leseprobe haben Sie beim  
 [edv-buchversand.de](http://edv-buchversand.de) heruntergeladen.  
Das Buch können Sie online in unserem  
Shop bestellen.

[Hier zum Shop](#)

## Vorwort

Alle, die sich mit dem Thema Softwareentwicklung beschäftigen, sei es in der Rolle eines Entwicklers, einer Architektin oder eines Projektleiters, kennen die Herausforderungen und Unwägbarkeiten, die stetig neue technologische Trends, Frameworks, Tools und Standards mit sich bringen. Die treibende Kraft hinter den Bemühungen ist häufig, dass die Entwicklung betrieblicher Anwendungssoftware weiter vereinfacht und damit effizienter gestaltet werden soll. Dabei ist allerdings auch zu berücksichtigen, dass sich bestehende Software und die damit getätigten Investitionen gut an die neuen Trends und Standards anpassen bzw. darin integrieren lassen sollen. Das ist in der Realität häufig nicht der Fall, denn oftmals werden technische Aspekte und fachliche Logik im Quellcode einer Anwendung vermischt. Das erschwert die Anpassbarkeit an veränderte technologische Rahmenbedingungen, wie etwa eine neue UI- oder Schnittstellentechnologie, und macht zum Teil sogar komplette Neuentwicklungen notwendig.

Während die Technik die eine Seite der Medaille der Softwareentwicklung ausmacht, steht auf der anderen Seite die vielfach zitierte Geschäftslogik. Über die Jahre haben wir immer wieder das folgende Credo gehört, das Sie sicherlich auch kennen:

*»Die Anwendungsentwicklung soll sich mehr auf ihre eigentliche Arbeit fokussieren, nämlich die Implementierung der Geschäftslogik auf Basis der Anforderungen der jeweiligen Fachbereiche.«*

Wird eine bessere Fokussierung gefordert, muss es wohl offensichtlich irgendeine Form der »Ablenkung« geben. Diese Ablenkung besteht z. B. in sich verändernden technologischen Rahmenbedingungen, die neben den fachlichen Anforderungen bei der Realisierung berücksichtigt werden müssen. Sie treiben typischerweise die Implementierungskosten (Total Cost of Development, kurz TCD) in die Höhe. Solche Ablenkungen gilt es also zu eliminieren.

Vor diesem Hintergrund ist in der Vergangenheit bereits der ABAP-Stack entstanden, ein reichhaltiges und stabiles technisches Fundament, auf dem die Anwendungsentwicklung schon seit vielen Jahren mit großem Erfolg aufsetzt. Das Business Object Processing Framework (BOPF) hat sich erstmalig der Standardisierung von Geschäftslogik, also der fachlichen Seite von Geschäftsanwendungen, angenommen. Das ABAP-Programmiermodell für SAP Fiori greift BOPF in einer CDS-basierten Variante auf und entkoppelt sich von technischen Komponenten, wie beispielsweise SAP Fiori Elements, durch die Verwendung von Annotationen.

Wenn wir also die Entwicklung von ABAP als Sprache zur Erstellung betrieblicher Anwendungssoftware mit den Grundideen von BOPF, den Programmiermodellen der ABAP-Plattform sowie dem Wunsch, evolutionsfähige Software zu bauen oder bestehende Software mit einem »Evolutionsschutz« auszustatten, vermengen, dann ist die logische Konsequenz ein im ABAP-Kern verankertes Programmiermodell. Es bringt eigene Repository-Objekte sowie umfangreiche Erweiterungen der ABAP-Sprache mit sich und sorgt von Haus aus dafür, dass Fachlichkeit und Technik zwei getrennte Seiten der gleichen Medaille bleiben. Da die Reise der Softwareentwicklung in die Cloud führt, bietet sich ein REST-basierter Ansatz an, um Anwendungsinteraktion von der genutzten Cloud-Infrastruktur zu entkoppeln. Als Ergebnis dieser Überlegungen ist das ABAP RESTful Application Programming Model zur Entwicklung von SAP-Fiori-Apps und Web-APIs entstanden.

Wir danken den Autoren und dem Verlag für die Erstellung dieses Buchs und wünschen allen Lesenden viel Freude damit.

**Volker Drees**  
RAP Product Expert

**Marcel Hermanns**  
RAP Product Lead/  
Chief Development Expert

## Einleitung

Seit jeher entwickeln sich die Programmiersprache ABAP und das zugehörige Programmiermodell stetig weiter. Mit dem aktuellen *ABAP RESTful Application Programming Model* (RAP) haben sich einschneidende Veränderungen in der Art und Weise, Anwendungen zu entwickeln, ergeben. Dieses Programmiermodell sieht einige komplett neue Programmierkonzepte, wie die Verhaltensdefinitionen und Business-Services, vor.

Das Ziel dieses Buchs ist es, Ihnen einen umfassenden Überblick über das neue Programmiermodell zu geben. Sie erwerben nicht nur das erforderliche theoretische Hintergrundwissen, sondern wenden das Erlernte auch in drei umfangreichen Praxiskapiteln an. Die dort beschriebenen praktischen Anwendungsfälle können Ihnen als Grundlage dienen, um die Vorgehensweise für Ihre eigenen Entwicklungsprojekte an das neue Programmiermodell anzupassen.

In Teil I, »Konzepte des ABAP RESTful Application Programming Model«, wird das Fundament für den anschließenden Praxisteil II, »Praktische Anwendungsentwicklung mit dem ABAP RESTful Application Model«, gelegt, und die grundlegenden Konzepte des Programmiermodells werden im Detail vorgestellt. Damit ist der erste Teil des Buchs für Leserinnen und Leser mit unterschiedlichen Vorkenntnissen geeignet. Er richtet sich sowohl an ABAP-Entwicklerinnen und -Entwickler, die bereits Erfahrungen mit dem Programmiermodell gesammelt haben und ihre Kenntnisse vertiefen wollen, als auch an solche, die bisher noch keine Erfahrungen mit den neuen Konzepten haben. Grundlegende Kenntnisse der ABAP-Entwicklung sollten jedoch bereits vorhanden sein. Für das Verständnis des Buchs ist es egal, ob Sie in einer Cloud-Umgebung oder auf einem SAP-S/4HANA-On-Premise-System.

Wenn Sie neu in das Thema einsteigen, bietet es sich an, das Buch vom ersten bis zum letzten Kapitel chronologisch durchzuarbeiten. Wenn Sie bereits erste Erfahrungen mit dem ABAP RESTful Application Programming Model gesammelt haben, kann es durchaus sinnvoll sein, wenn Sie einzelne Kapitel überspringen und gezielt tiefer in die für Sie interessantesten Themen eintauchen. Die Beispiele aus Teil II sind voneinander unabhängig und bauen nicht aufeinander auf.

Ziel dieses Buchs

Zielgruppe



### Programmierbeispiele zum Download

Alle Programmierbeispiele aus dem Buch stehen auf der Seite [www.sap-press.de/5412](http://www.sap-press.de/5412) auf der Registerkarte **Materialien** zur Verfügung. Von einigen Entwicklungsobjekten der Beispieldatenmodelle und -anwendungen werden in den Listings dieses Buchs nur Ausschnitte abgedruckt. In diesem Fall finden Sie den vollständigen Quelltext ebenfalls im Download-Bereich.

**Aufbau dieses Buchs** Wie bereits erwähnt, besteht das Buch aus zwei Teilen. Im ersten Teil I lernen Sie die grundlegenden Konzepte des ABAP RESTful Application Programming Model kennen.

**Teil I** In **Kapitel 1**, »Einführung in das ABAP RESTful Application Programming Model«, stellen wir das Programmiermodell vor und ordnen seine Softwarearchitektur in andere Architekturmodelle ein. Die verschiedenen Anwendungsfälle und Varianten der RAP-Entwicklung werden hier ebenfalls erläutert.

In **Kapitel 2**, »Core Data Services: Datenmodellierung«, werden die Möglichkeiten und Funktionen von CDS beschrieben und anhand von Beispielen erläutert. Da es sich bei CDS um einen zentralen Baustein des Programmiermodells handelt, bildet dieses Kapitel die Grundlage für die darauf aufbauenden Sprachelemente und Entwicklungsobjekte.

**Kapitel 3**, »Verhaltensdefinition«, erläutert aufbauend auf der Einführung in CDS die verschiedenen Implementierungstypen des Programmiermodells. Wir beschreiben den Aufbau, die wesentlichen Sprachelemente und die Möglichkeiten von Verhaltensdefinitionen anhand zahlreicher Beispiele.

In **Kapitel 4**, »Entity Manipulation Language: Zugriff auf Geschäftsobjekte«, werden das Konzept, die Syntax und die Möglichkeiten der in ABAP integrierten EML beschrieben. An dieser Stelle schlagen wir auch bereits die Brücke zur Verhaltensimplementierung.

**Kapitel 5**, »Verhaltensimplementierung«, erläutert im Detail, wie das Interface zur Implementierung der Geschäftslogik einer Anwendung aufgebaut ist und an welchen Stellen und zu welchen Zeitpunkten eine Implementierung von Verhalten umgesetzt werden kann.

In **Kapitel 6**, »Business-Services«, zeigen wir Ihnen, wie Sie mithilfe einer Servicedefinition den Umfang eines OData-Service für ein Geschäftsobjekt festlegen können und wie Sie mit dem Service-Binding das Protokoll des Service festlegen.

**Kapitel 7**, »Anwendungsoberflächen und SAP Fiori Elements«, richtet den Blick auf die Benutzeroberflächen Ihrer RAP-Anwendungen. Hier gehen wir nicht in der Tiefe auf alle Funktionen von SAP Fiori ein, sondern zeigen Ihnen, wie Sie mit SAP Fiori Elements einfache Oberflächen generieren und mit Annotationen näher konfigurieren können. Damit möchten wir vor allem die Grundlage für die folgenden Praxiskapitel schaffen.

In Teil II werden die verschiedenen Entwicklungsszenarien des ABAP RESTful Programming Model jeweils anhand eines konkreten Beispiels veranschaulicht.

Teil II

Zunächst betrachten wir dazu in **Kapitel 8**, »Anwendungsfälle«, die unterschiedlichen Anwendungsfälle des Programmiermodells in der ABAP-Entwicklung und diskutieren die Eignung der verschiedenen RAP-Szenarien im jeweiligen Kontext. Das Kapitel beantwortet damit die Frage, wann welcher Implementierungstyp eingesetzt werden soll.

Ihre erste Anwendung auf Basis des ABAP RESTful Application Programming Model erstellen Sie in **Kapitel 9**, »Managed Scenario: Entwicklung einer Anwendung mit SAP Fiori Elements«. Dabei handelt es sich um eine komplett eigenständige Anwendung ohne Bezug zum SAP-Standard. Im Fokus steht hier die Umsetzung des Managed Scenario. Die Anwendung wird im Rahmen der Verhaltensdefinition und -implementierung mit Aktionen, Validierungen und Ermittlungen angereichert.

Auch in **Kapitel 10**, »Managed Scenario mit Unmanaged Save: Bestehende Anwendung integrieren«, befassen wir uns mit dem Managed Scenario, implementieren allerdings eine eigene Verbuchungslogik. Themen wie die Nummernvergabe oder die Implementierung der Speichersequenz sind hier zentral. Ziel dieser zweiten Anwendung ist es, SAP-Standardfunktionen in die Eigenentwicklung einzubinden.

In **Kapitel 11**, »Unmanaged Scenario: Vorhandenen Quellcode wiederverwenden«, gehen wir von einer bereits bestehenden kundeneigenen Anwendung aus, die im Rahmen des Unmanaged Scenario mit dem ABAP RESTful Application Programming Model abgebildet wird. Ein wesentlicher Punkt in diesem Kapitel ist die Implementierung der Interaktionsphase, da diese beim Unmanaged Scenario im Gegensatz zum Managed Scenario nicht vom RAP-Framework zur Verfügung gestellt wird.

In **Kapitel 12**, »Besonderheiten im SAP BTP, ABAP Environment«, stellen wir die Unterschiede zwischen der Cloud-Entwicklung und der On-Premise-Entwicklung heraus. Sie erfahren auch, wie Sie eine RAP-Anwendung an die technischen Gegebenheiten des ABAP Environment anpassen können.

**Kapitel 13**, »Ausblick«, wirft einen kurzen Blick auf die geplante Weiterentwicklung des Programmiermodells, beispielsweise hinsichtlich der Erweiterbarkeit, der Integration und Wiederverwendbarkeit und seiner Rolle im Umfeld von SAP S/4HANA Cloud.

**Informationskästen** In hervorgehobenen Informationskästen finden Sie in diesem Buch Inhalte, die wissenswert und hilfreich sind, aber etwas außerhalb der eigentlichen Erläuterung stehen. Damit Sie diese Informationen sofort einordnen können, haben wir die Kästen mit entsprechenden Symbolen gekennzeichnet:

**[>>]** In Kästen, die mit diesem Symbol gekennzeichnet sind, finden Sie Informationen zu *weiterführenden Themen*.

**[!]** Dieses Symbol weist Sie auf *Besonderheiten* hin, die Sie beachten sollten. Es *warnt Sie* außerdem vor häufig gemachten Fehlern oder Problemen, die auftreten können.

**[+]** Die mit diesem Symbol gekennzeichneten *Tipps* geben Ihnen spezielle Empfehlungen aus unserer Projektpraxis, die Ihnen die Arbeit erleichtern können.

**[zB]** *Beispiele*, durch dieses Symbol kenntlich gemacht, weisen auf Szenarien aus der Praxis hin und veranschaulichen die dargestellten Funktionen.

### Danksagungen

Wir möchten uns für die freundliche Unterstützung und die Bereitstellung der Infrastruktur durch die All for One Group SE bedanken, ohne die dieses Buch nicht möglich gewesen wäre. Insbesondere bedanken wir uns bei Lars Woll und Andreas Klose.

Ein besonderer Dank geht außerdem an die Kollegen bei SAP für die tatkräftige Unterstützung und das konstruktive Feedback. Hier möchten wir speziell Marcel Hermanns und Volker Drees hervorheben.

Vielen Dank auch an Janina Schweitzer von SAP PRESS für die sehr gute Zusammenarbeit und kompetente Unterstützung im Rahmen des Lektorats.

Ein individueller Dank geht ...

- von Michael Lensch an seine Frau Stefanie und den gemeinsamen Sohn Paul für die Unterstützung und das Verständnis dafür, dass er unzählige Stunden und viele Wochenenden in das Buch investiert hat.
- von Matthias Jäger an seine Partnerin Andrea und seine beiden Kinder Emilia und Clemens, die ihm geduldig und unterstützend Raum und Zeit für dieses Buch gegeben haben.

Wir hoffen, dass Ihnen dieses Buch als hilfreiches Nachschlagewerk und Leitfaden für die Entwicklung von Anwendungen auf Basis des ABAP RESTful Application Programming Model dienen wird und Ihnen den Einstieg in das neue Programmiermodell erleichtert. Viel Vergnügen beim Lesen!

**Lutz Baumbusch, Matthias Jäger und  
Michael Lensch**