

Einleitung

ABAP ist die von SAP entwickelte Programmiersprache für die Erstellung kundenspezifischer Anwendungsprogramme. Die Abkürzung steht für Advanced Business Application Programming. Darüber hinaus nutzt SAP selbst ABAP auch als Basis für SAP ERP, die SAP Business Suite und die SAP-NetWeaver-Plattform. All diese Systeme basieren auf Millionen von in ABAP geschriebenen Zeilen.

Die Abkürzung ABAP bedeutete ursprünglich Allgemeiner Berichtsaufbereitungsprozessor und war, wie es der Name vermuten lässt, dafür gedacht, kundeneigene Berichte aufzubereiten. ABAP orientierte sich in seiner ursprünglichen Form an der Syntax der Programmiersprache COBOL. Über die Jahrzehnte wurde ABAP kontinuierlich weiterentwickelt und verbessert. So hat sich der Sprachumfang mittlerweile vervielfacht, da immer wieder neue Anweisungen und Funktionen hinzukamen, wobei ältere aus Kompatibilitätsgründen beibehalten wurden.

Die aktuelle Version von ABAP (7.54, Stand Ende 2020) steht in ihrer Funktionalität anderen modernen Programmiersprachen kaum noch nach. Bereits 2004 wurde der Sprachumfang um die objektorientierten Elemente von *ABAP Objects* erweitert. Dennoch findet man aufgrund der erwähnten Kompatibilitätssicherung in ABAP auch heute noch viel ältere Funktionalität, die sich mit den neuen, modernen Konstrukten vermischt hat. In der Praxis hat es der Entwickler daher häufig mit einer Mischform aus »alter« und »neuer« Welt zu tun. Eine einseitige Betrachtung ergibt daher keinen Sinn.

Dennoch ist die Zukunft von ABAP bereits angebrochen. ABAP hat durch die letzten beiden Releases 7.40 und 7.50/7.51 sehr viele neue Funktionen gewonnen, die in die Kapitel dieses Buches integriert sind. In Abbildung 1 können Sie die verschiedenen ABAP-Versionen über die Jahre dargestellt sehen.

Auffällig ist hier, dass zwischen 2004 und 2013 praktisch keine Neuerungen in ABAP eingeführt wurden. Dies war auch die Zeit, als es hieß, dass Java der Nachfolger von ABAP werden sollte. Dies hat sich aber zum Glück nicht durchgesetzt.

Besonders wichtig in Bezug auf dieses neue ABAP sind die neue Datenbank SAP HANA und die neue Oberflächentechnologie SAPUI5 bzw. die von SAP bereitgestellten SAP-Fiori-Apps. Diese neuen Technologien sind das Aushängeschild des SAP-ERP-Nachfolgers SAP S/4HANA. Doch auch wenn die SAP Business Suite für SAP S/4HANA (auf Vorlage der alten Systeme) komplett überarbeitet wurde, bleibt ABAP die Kernprogrammiersprache auch für dieses System.

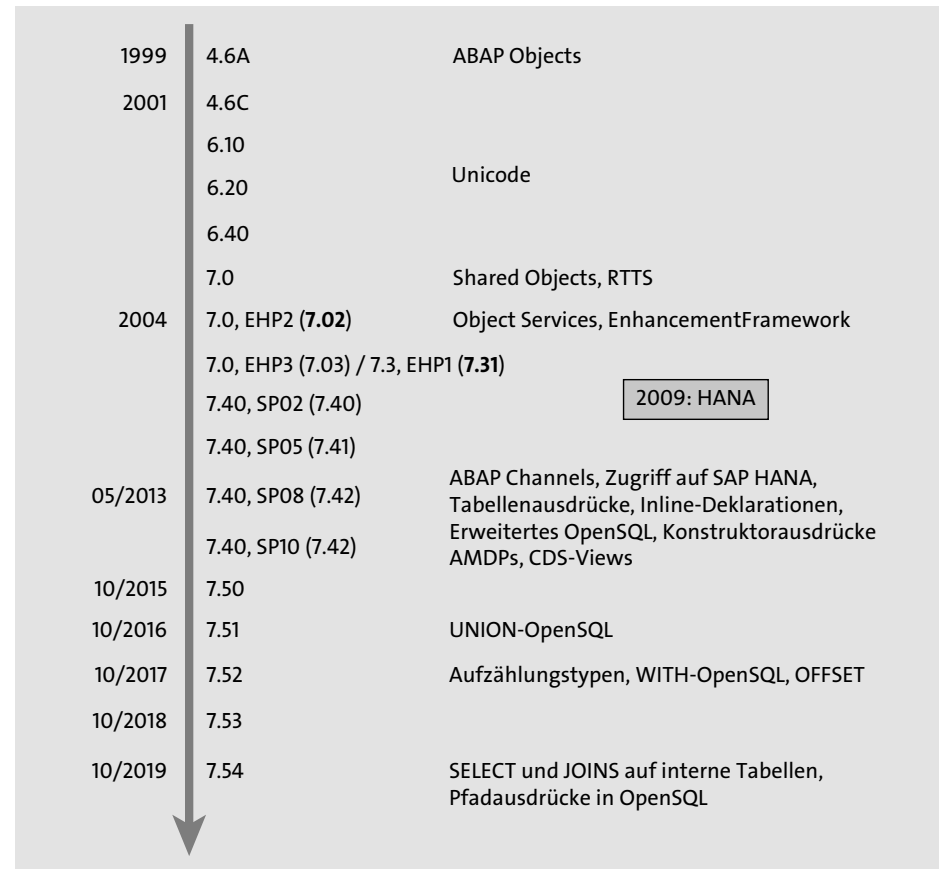


Abbildung 1 Neuerung in ABAP seit 1999 bis heute



Altlasten existieren weiterhin

Es ist wichtig zu verstehen, dass in der On-Premise-Version von SAP S/4HANA weiterhin alle alten Sprachelemente vorzufinden sind. Das heißt, Sie können auch hier Makros, Unterprogramme, Funktionsbausteine etc. verwenden. Technische gesehen ist es für uns ABAP-Entwickler der gleiche Unterbau. In anderen Worten: Es muss alles beherrscht werden, insbesondere, wenn Altanwendungen gewartet oder erweitert werden sollen.

Betrachtet man die durchschnittliche Halbwertszeit (d. h. die Einsatzzeit beim Kunden) eines ERP-Systems von ca. 20 Jahren (SAP R/3 kam 1993, SAP ECC 2004), wird uns ABAP durch SAP S/4HANA noch lange erhalten bleiben – auch wenn dies in den letzten Jahren (z. B. im Rahmen der Ankündigung der Java-Strategie) nicht immer als Selbstverständlichkeit galt.

In anderen Worten: ABAP ist die Programmiersprache von SAP und wird dies bleiben. Herzlich willkommen seien daher alle Leser, die sich entschieden haben, sich intensiver mit ABAP im Allgemeinen und ABAP Objects im Speziellen zu beschäftigen.

Aufbau dieses Buches

Zuallererst möchte ich Ihnen einen Überblick über das vor Ihnen liegende Buch geben. Im Anschluss habe ich Ihnen verschiedene Lesepläne zusammengestellt – abhängig davon, welche Vorkenntnisse Sie mitbringen.

- **Teil I, »Die Werkzeugkiste des ABAP-Entwicklers«**, beschäftigt sich mit den Entwicklungswerkzeugen des SAP GUI wie der eingebauten ABAP Workbench (Transaktion SE80). Außerdem stelle ich Ihnen hier die neuen ABAP Development Tools for SAP NetWeaver (ADT) für die Entwicklungsumgebung Eclipse vor.
- **Teil II, »Der Kern der Sprache ABAP«**, beschäftigt sich mit dem ABAP-Sprachkern. Von absolut grundlegenden Anweisungen über den Zugriff auf Datenbanktabellen, die Verarbeitung der Daten und ABAP-Objects-Anweisungen bis hin zur Report- und Dynpro-Programmierung ist hier alles vertreten. So hat hier auch der Zugriff auf die neue SAP-HANA-Datenbank mit ABAP (auch *ABAP für HANA* genannt) mit Eingang gefunden. Insbesondere finden Sie in diesem Teil die mit den Releases 7.40 und 7.50/1 neu eingeführten Sprachbefehle und Funktionen.
- **Teil III, »Techniken zur Qualitätssicherung«**, dreht sich um den Test und die Qualitätskontrolle Ihrer geschriebenen Anweisungen. Hier werden die gezielte Verwendung des ABAP Debuggers sowie die zur Verfügung stehenden Analysewerkzeuge erläutert.
- **Teil IV, »Fortgeschrittene Programmieretechniken«**, behandelt weitere Themen aus dem Umfeld von ABAP. Dazu gehören die Entwicklung von Schnittstellen, die Erweiterung von SAP-Standardprogrammen, die Entwicklung von Formularen und Tabellenanzeigen sowie auch fortgeschrittene Techniken wie die dynamische Programmierung oder die Anwendung objektorientierter Frameworks wie der Object Services oder Shared Objects.
- **Teil V, »Objektorientierte Programmierung«**, vermittelt in erster Linie die Grundlagen der objektorientierten Programmierung mit ABAP und zeigt konkrete Methoden für den Entwicklungsprozess. Insbesondere werden hier die objektorientierte Modellierung sowie Hilfsmittel wie UML und Entwurfsmuster besprochen.
- Im letzten Teil des Buches, **Teil VI, »Ein Blick über den Tellerrand: Was Sie als ABAP-Entwickler sonst noch kennen sollten«**, möchte ich Ihnen den Grund für die vielen Änderungen der letzten Releases etwas genauer vorstellen: SAP HANA als neue Datenbankplattform sowie SAPUI5 als neue Web-Frontend-Technologie, die eine regelrechte Innovationswelle ab 2010 angestoßen haben. In diesem Teil

gehe ich insbesondere sowohl auf die vielen neuen Begriffe im Umfeld dieser Technologien als auch auf die Entwicklung von OData-Services ein.

Der Anhang fasst alle wichtigen ABAP-Anweisungen, obsoleten ABAP-Anweisungen, Systemfelder, eingebauten Datentypen, nützliche Funktionsbausteine und Klassen, Transaktionscodes sowie technische Tabellen zur Programmierung mit ABAP zusammen.

Ihr Weg durch das Buch


Da sich dieses Buch an mehrere Zielgruppen richtet (ABAP-Einsteiger, -Kenner und -Experten) und gleichermaßen als Nachschlagewerk für den Alltag dienen soll, finden Sie im Folgenden einige Hinweise, wie Sie aus dem Buch für sich am meisten Gewinn erzielen können.


Aus folgenden Leseplänen können Sie wählen:


- Einsteiger in ABAP
- Einsteiger in ABAP Objects
- Kenner von ABAP Objects (vor allem der Releases vor und bis 7.31)
- Leser der Voraufgabe
- Interessenten an den neuen Technologien
- Experten

Die Lesepläne bauen aufeinander auf, vom Einsteiger bis zum Experten. Versuchen Sie, wann immer es geht, so viel wie möglich am System nachzuvollziehen, da sich Programmierung am besten durch die konkrete Anwendung erlernen lässt.

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

 Die mit diesem Symbol gekennzeichneten *Tipps* geben Ihnen spezielle Empfehlungen, die Ihnen die Arbeit erleichtern können.

 In Kästen, die mit diesem Symbol gekennzeichnet sind, finden Sie Informationen zu *weiterführenden Themen* oder wichtigen Inhalten, die Sie sich merken sollten.

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

Leseplan für Einsteiger in ABAP

Fangen Sie mit der Lektüre des Kapitel 1 an, um sich mit der Entwicklungsumgebung vertraut zu machen. Insbesondere ist hier Abschnitt 1.4.2, »Einen Report anlegen«,

von Bedeutung, der Ihnen zeigt, wie Sie einen neuen Report in Transaktion SE80 anlegen und ausführen können.

Bevor Sie sich dann mit den Anweisungen von ABAP auseinandersetzen, ist es wichtig, die Rolle des in Kapitel 3 beschriebenen ABAP Dictionarys zu verstehen. Lesen Sie sich hier alle Abschnitte bis Abschnitt 3.4, »Tabellentypen«, durch. Die folgenden Abschnitte können Sie erst mal außer Acht lassen.

Mit Ihrem ersten angelegten Report und dem Grundwissen über das ABAP Dictionary können Sie nun die in Kapitel 7, »Die ABAP-Grundbefehle«, beschriebenen grundlegenden ABAP-Anweisungen ausprobieren. Versuchen Sie, die vielen Beispiele direkt bei sich im System auszuprobieren. Überspringen Sie erst mal die als fortgeschritten markierten Abschnitte.

Nach einer kurzen Verschnaufpause können Sie sich direkt intensiv mit Kapitel 8 auseinandersetzen, das die internen Tabellen von ABAP intensiv behandelt. Kapitel 9 zeigt Ihnen im Anschluss, wie Sie Daten von der Datenbank in interne Tabellen laden und weiterverarbeiten können.

Lesen Sie nun die restlichen Abschnitte (ab Abschnitt 3.5) von Kapitel 3, »Das ABAP Dictionary«. Insbesondere ist hier Abschnitt 3.5, »Datenbanktabellen«, von Bedeutung, der Ihnen zeigt, wie Sie selbst eine Tabelle auf der Datenbank anlegen können. So eine Tabelle können Sie mit den in Kapitel 9, »Zugriff auf Datenbanken«, gelernten Open-SQL-Anweisungen bzw. den in Abschnitt 3.9 beschriebenen Pflegedialogen bearbeiten.

Haben Sie eine SAP-HANA-Datenbank zur Verfügung, empfehle ich Ihnen nun die Lektüre von Kapitel 30, »SAP HANA«, sowie anschließend von Kapitel 10, »Zugriff auf SAP-HANA-Entwicklungsobjekte«. Sollten Sie keine SAP-HANA-Datenbank zur Verfügung haben, so ist dies nicht weiter schlimm, da Sie diese beiden Kapitel jederzeit nachträglich ohne Probleme durcharbeiten können.

Kapitel 12 beschäftigt sich dann mit der Erstellung von Selektionsbildschirmen in Reports. Dies hilft Ihnen dabei, Eingabemasken für Ihre Anwender bereitzustellen. Kapitel 4 zeigt Ihnen, wie Sie es schaffen, Ihren Report mit einem Transaktionscode zu verknüpfen, damit Ihre Anwender den von Ihnen entwickelten Report direkt aufrufen können.

Da die Programme durch die vielen Anweisungen immer größer werden, finden Sie in Kapitel 13 die in ABAP zur Verfügung stehenden Strukturierungselemente, die Ihnen dabei helfen, Ihre Programme in kleinere, voneinander abgetrennte Bereiche zu unterteilen. Insbesondere ist hier Abschnitt 13.4, »Funktionsbausteine«, von Bedeutung. Lesen Sie sich dazu auch Kapitel 5, »Der Function Builder«, durch, um zu verstehen, wie Sie eigene Funktionsbausteine anlegen und verwenden können. Natürlich können Sie auch direkt Klassen und Interfaces zur objektorientierten Strukturierung Ihrer Programme nutzen, wie in Kapitel 11, »Die ABAP-Objects-Syntax«, beschrieben.

Anschließend können Sie dem Leseplan für Einsteiger in ABAP Objects folgen. Haben Sie bis hierhin alle Kapitel gemeistert, können Sie sich dem Umfeldwissen zu ABAP widmen, hierzu gehören die Kapitel in Teil III, Teil IV und Teil VI. Heben Sie sich Kapitel 24, »Fortgeschrittene Programmieretechniken«, bis ganz zum Schluss auf.

Um das große Ganze zwischen all den ABAP-Technologien zu verstehen, empfehle ich Ihnen außerdem Kapitel 29, »ABAP-Programmiermodelle«, wo ich Ihnen einen Überblick über die aktuelle und zukünftige Art der Programmierung mit ABAP erläutere.

Leseplan für Einsteiger in ABAP Objects

Dieser Leseplan ist für all jene gedacht, die sich mit dem Sprachkern von ABAP bestens auskennen, aber noch nicht die Chance hatten, sich mit ABAP Objects auseinanderzusetzen, oder bei denen der letzte Einsatz der objektorientierten Konzepte schon eine Weile her ist.

Lesen Sie zunächst Kapitel 25, »Grundlagen der Objektorientierung«, sowie Kapitel 11, »Die ABAP-Objects-Syntax«. Üben Sie nach der Lektüre dieser beiden Kapitel erst einmal intensiv die Anwendung von Klassen und Instanzen am Beispiel des SAP List Viewers (ALV) mithilfe von Kapitel 19. Konzentrieren Sie sich hier insbesondere auf die neue ALV-Anzeige auf Basis der Klasse `CL_SALV_TABLE`, die in Abschnitt 19.2 beschrieben ist.

Legen Sie, wie in Kapitel 6, »Der Class Builder«, beschrieben, eine eigene Klasse an, die Sie anschließend in Ihrem Report instanziiieren und verwenden können. Hinweise zur objektorientierten Gestaltung von Reports gibt Abschnitt 28.6, »Objektorientierte Reports«.

Arbeiten Sie nun schrittweise die restlichen Kapitel in Teil V durch. Am Ende in der Einleitung zu Teil V finden Sie Verweise auf die konkrete Anwendung bereits in SAP existierender objektorientierter Frameworks. Dazu gehört beispielsweise das Business Object Processing Framework, das Sie in Kapitel 23 beschrieben finden.

Leseplan für Kenner von ABAP Objects

Sie kennen bereits ABAP und ABAP Objects und sind vor allem an den Neuerungen bis Release 7.54 interessiert? Dann ist dieser Leseplan richtig für Sie.

Wenn Sie die ABAP Development Tools (ADT) für Eclipse noch nicht kennen, sollten Sie direkt mit Kapitel 2 einsteigen. Anschließend sind für Sie insbesondere die Übersichten in den folgenden Abschnitten interessant, die Auskunft über die ABAP-Neuerungen geben und jeweils auf die Abschnitte im Buch verweisen, in denen diese ausführlicher behandelt werden:

- Abschnitt 7.15, »Änderungen und Neuerungen bis ABAP 7.54«
- Abschnitt 8.12, »Änderungen und Neuerungen im Umfeld von internen Tabellen bis ABAP 7.54«
- Abschnitt 9.9, »Änderungen und Neuerungen im Umfeld von Open SQL bis ABAP 7.54«
- Abschnitt 10.4, »Änderungen und Neuerungen beim Zugriff auf SAP-HANA-Entwicklungsobjekte bis ABAP 7.54«
- Abschnitt 11.15, »Änderungen und Neuerungen in ABAP Objects bis ABAP 7.54«

Insbesondere sollten Sie nach der Lektüre folgende Begriffe verstanden haben und anwenden können:

- Inline-Deklaration
- Tabellenausdrücke
- Änderungen an Open SQL, z. B. die neuen Open-SQL-Ausdrücke, eine Selektion mit `SELECT` auf interne Tabellen oder die Verwendung der neuen Anweisung `WITH`
- Konstruktorausdrücke, insbesondere die Verwendung der Zusätze von `BASE` und `LET`
- den Zusatz `GROUP BY` der Anweisung `LOOP AT`
- ABAP Managed Database Procedures (AMDP) zur Programmierung von Stored Procedures in Klassen (für SAP HANA)
- AMDP BAdIs
- ABAP-CDS-Views
- Mesh-Typen

Leser der Voraufgabe

Sie kennen bereits die erste Auflage dieses Buches und sind nur am Delta von ABAP 7.31 und ABAP 7.54 interessiert?

Dann sind für Sie die folgenden Abschnitte interessant, die Auskunft über die ABAP-Neuerungen geben und jeweils auf die Abschnitte im Buch verweisen, in denen diese ausführlicher behandelt werden.

- Abschnitt 7.15, »Änderungen und Neuerungen bis ABAP 7.54«
- Abschnitt 8.12, »Änderungen und Neuerungen im Umfeld von internen Tabellen bis ABAP 7.54«
- Abschnitt 9.9, »Änderungen und Neuerungen im Umfeld von Open SQL bis ABAP 7.54«
- Abschnitt 10.4, »Änderungen und Neuerungen beim Zugriff auf SAP-HANA-Entwicklungsobjekte bis ABAP 7.54«
- Abschnitt 11.15, »Änderungen und Neuerungen in ABAP Objects bis ABAP 7.54«

Zusätzlich sind zwei neue Kapitel zum Buch hinzugekommen, deren Lektüre ich Ihnen empfehle:

- Kapitel 23, »Business Object Processing Framework«
- Kapitel 29, »ABAP-Programmiermodelle«

Interessant ist hier vor allem Kapitel 29, weil es aufzeigt, wie sich SAP die Entwicklung mit ABAP in den nächsten zehn Jahren vorstellt.

Leseplan für Interessenten an den neuen Technologien

Um einen Überblick über SAP HANA und die neue Web-Frontend-Technologie zu erhalten, ist insbesondere Teil VI mit Kapitel 30, »SAP HANA«, und Kapitel 31, »SAPUI5, SAP Fiori und SAP Gateway«, von Bedeutung. In Kapitel 31 wird der Fokus auf die Entwicklung von OData-Services mit ABAP gelegt. Kapitel 10, »Zugriff auf SAP-HANA-Entwicklungsobjekte«, fasst die ABAP-Anweisungen zusammen, die Sie für den Zugriff auf die SAP-HANA-Datenbank benötigen. Vor allem aber mit den in Kapitel 2, »Die ABAP Development Tools«, beschriebenen ADT als Nachfolger der ABAP Workbench sollten Sie sich vertraut machen.

Leseplan für Experten

Sie wissen bereits, welche sprachlichen Neuerungen es bis Release 7.54 gibt, und haben diese Anweisungen auch bereits verinnerlicht? Dann sind Sie hier richtig!

Informieren Sie sich z. B. über den neuen SAP List Viewer mit integriertem Datenzugriff (ALV with Integrated Data Access, IDA, siehe Abschnitt 19.3). Informationen zur Anwendung von Mesh-Typen finden Sie in Abschnitt 8.11, »Meshes«.

Kapitel 24 behandelt Themen der fortgeschrittenen Programmierung, insbesondere Parallelisierung, dynamische Programmierung und dynamisches SQL. Außerdem finden Sie hier Informationen zu persistenten Klassen und Shared Objects sowie der Arbeit mit XML und JSON. Kapitel 28 zeigt Ihnen, wie Sie Entwurfsmuster auf ABAP Objects anwenden können. Gerade die Implementierung des Model-View-Controllers und des Observer-Musters wird Sie hier interessieren. Zusätzlich lohnt sich die Kenntnis des Business Object Processing Frameworks (BOPF), das Sie in Kapitel 23 beschrieben finden.

Ihr individueller Leseweg

Abweichend von den hier vorgestellten Leseplänen können Sie sich dem Buch natürlich auf vielen unterschiedlichen Wegen nähern. Da das Buch eine breite Zielgruppe ansprechen und gerade für den Alltag nützlich sein soll, habe ich versucht, mehrere Zugänge zum Buch zu ermöglichen. Stöbern Sie doch einfach einmal durch das Inhaltsverzeichnis, oder suchen Sie nach den Themen, die Sie interessieren im Index.

Im Anhang finden Sie außerdem ein Glossar mit den wichtigsten Fachbegriffen sowie nützliche Übersichten mit ABAP-Anweisungen, wichtigen Klassen und Funktionsbausteinen, Transaktionscodes, technischen Tabellen und vieles mehr.

Bleibt mir nur noch, Ihnen viel Erfolg und Spaß bei der Arbeit mit dem Buch zu wünschen. Ich hoffe, es hilft Ihnen dabei, die kleinen und großen Herausforderungen des Alltags zu meistern. Die Zusammenstellung des Inhalts basiert vor allem auf meiner Tätigkeit als ABAP-Entwickler und meinen Erfahrungen als Trainer. Mein Dank gilt daher insbesondere denjenigen Teilnehmern meiner Kurse, die immer eine Frage mehr gestellt und mir auf diese Weise neue Sichtweisen eröffnet haben. So gesehen ist das vorliegende Buch ein Gemeinschaftswerk von und für (angehende) ABAPer. Ich freue mich daher auch über jeden Kommentar, der dazu beiträgt, das Buch weiter zu verbessern und abzurunden. Schreiben Sie mir gerne unter der E-Mail-Adresse abap.felix.roth@gmail.com.

In diesem Sinne: COMMIT WORK!

Felix Roth