

Einleitung

Die Entwicklung mit ABAP hat sich seit Einführung mit SAP R/2 in den 1980er Jahren im Laufe der letzten 30 Jahre stetig weiterentwickelt. Ursprünglich stand die Abkürzung ABAP einmal für **Allgemeiner Berichtsaufbereitungsprozessor** und enthielt nur Befehle zur Datenauswertung. Der SAP-NetWeaver-Applikationsserver ABAP bietet mittlerweile eine vollwertige Entwicklungsplattform, die die Basis für die meisten SAP-Standardanwendungen sowie eine große Zahl von Partnerlösungen und Kundenentwicklungen bildet. Einhergehend mit dem gewachsenen Funktionsumfang der Programmiersprache, der Serverinfrastruktur und in der Konsequenz der Geschäftsanwendungen wurden auch die Ansprüche an die Qualität, Sicherheit und Performance der hiermit erstellten Software größer.

Motivation

Gerade bei geschäftskritischen Anwendungen spielt die Qualitätssicherung mithilfe von Tests eine wichtige Rolle. Da ABAP-Programme zudem oft über einen langen Zeitraum im Einsatz sind, sollten die Entwicklungen gut verständlich und leicht anpassbar sein. Das Thema Performance hatte schon immer einen großen Stellenwert. Kurze Antwortzeiten und ein hoher Durchsatz waren und sind auch heute noch sehr wichtig. Durch die gestiegenen Datenvolumina und Benutzerzahlen von heute ergeben sich allerdings neue Herausforderungen, die nicht ausschließlich durch den Einsatz leistungsfähigerer Hardware beantwortet werden können. Anwendungen müssen weiterhin effizient mit den vorhandenen Systemressourcen umgehen, um skalierbar und schnell zu sein. In Zeiten, in denen ABAP-Programme nur von wenigen Personen im Unternehmen ausgeführt wurden, hatte das Thema Sicherheit noch einen relativ geringen Stellenwert. Heute sind ABAP-Anwendungen stark vernetzt und können insbesondere auch über das Internet von einem großen Nutzerkreis verwendet werden, müssen deshalb also viel höheren Sicherheitsanforderungen gerecht werden. Erfolgreiche Angriffe können einen immensen Schaden verursachen.

Qualität, Performance und Sicherheit

Aufbau des Buches

Moderne Anwendungsentwicklung mit ABAP erfordert also eine holistische Berücksichtigung der Themen Qualität, Performance und

Sicherheit. Bisher gab es Bücher, die sich mit einzelnen dieser Themen befasst haben. In diesem Buch behandeln wir die Themen zusammen, weil sie *alle* Teil professioneller Softwareentwicklung sind. So kann beispielsweise eine Anwendung, die zwar sicher und robust, aber nicht schnell genug läuft, zu Problemen im Produktivbetrieb führen. Ebenso wird Software, die zwar schnell und sicher ist, sich aber schlecht testen lässt, im Fall von Änderungen hohe Wartungskosten nach sich ziehen. Und schließlich ist Software, die zwar schnell und wartungsfreundlich ist, aber nicht sicher, ein Sicherheitsrisiko für jede Firma. Anhand dieser einfachen Beispiele wird deutlich, dass sich diese Themen nicht isoliert behandeln lassen. Aus diesem Grund möchten wir Ihnen die wichtigsten Empfehlungen, die jeder ABAP-Entwickler kennen sollte, in einem Kontext präsentieren. Dieses Buch ist in fünf Teile gegliedert, die Sie auch unabhängig voneinander lesen können.

Teil I: Qualität bei
ABAP-Entwick-
lungen

Im ersten Teil des Buches wollen wir Ihnen einen Einstieg in die Thematik geben, die Sie bei der Lektüre der späteren Kapitel unterstützt. Wir beginnen in **Kapitel 1**, »Qualität, Performance und Sicherheit in der aktuellen Softwareentwicklung«, mit einer allgemeinen Einführung zu Softwarequalität im Kontext von ABAP-Anwendungen und stellen dabei die typischen Entwicklungsszenarien vor. Danach steigen wir in **Kapitel 2**, »Anwendungsentwicklung mit ABAP in der Praxis«, zunächst in den technischen Aufbau eines ABAP-Systems ein und gehen auf die aus Qualitätssicht wichtigsten Elemente der ABAP-Programmierung ein. Im Anschluss nennen wir empfohlene Komponenten, die Sie bei einer modernen Anwendungsentwicklung unterstützen, sowie einige zusätzliche Produkte oder Dienstleistungen, von denen Sie im Rahmen einer Qualitätssicherung profitieren können.

Teil II: Guten
ABAP-Code erstel-
len und testen

Im zweiten Teil befassen wir uns mit dem Thema der qualitativen ABAP-Programmierung. In **Kapitel 3**, »ABAP-Codequalität«, erhalten Sie zunächst eine Einführung in Clean Code Development; anschließend stellen wir Ihnen Best Practices wie ABAP Objects, das Arbeiten mit Interfaces und weitere Methoden und Techniken zur Qualitätssteigerung Ihrer ABAP-Anwendungen vor. Wie Sie Unit-Tests in ABAP entwickeln, zeigen wir Ihnen in **Kapitel 4**, »Modultests mit ABAP Unit«. Dort erfahren Sie mehr über die Entwicklung von testbarer Software und den Einsatz von parametrisierten Unit-Tests. In **Kapitel 5**, »Werkzeuge für die Qualitätssicherung«, erhalten Sie eine Einführung in die Werkzeuge ABAP Test Cockpit (ATC), SAP Code

Inspector und ABAP in Eclipse (AiE). Welche Möglichkeiten Ihnen Vorgehensmodelle und Best Practices bieten, um die Basis für eine qualitativ hochwertige Softwareerstellung zu schaffen, erfahren Sie in **Kapitel 6**, »Prozesse und Methoden«.

Im dritten Teil zeigen wir Ihnen, wie Sie die Performance Ihrer ABAP-Anwendungen analysieren und optimieren können. In **Kapitel 7**, »Typische Performanceprobleme und Lösungen«, lernen Sie typische Performanceprobleme kennen. Wir zeigen Ihnen, wie Sie diese bei Ihren ABAP-Eigenentwicklungen vermeiden können. Die hierfür nützlichen Werkzeuge stellen wir Ihnen in **Kapitel 8**, »Werkzeuge für die Performanceanalyse«, vor. In **Kapitel 9**, »Durchführung von Performanceanalysen«, zeigen wir Ihnen, wie Sie bei der Performanceanalyse vorgehen können. Sie lernen auch, wie Sie die Werkzeuge am besten einsetzen. Am Ende des dritten Teils geben wir Ihnen in **Kapitel 10**, »Neue Möglichkeiten zur Performanceoptimierung mit SAP HANA und Open SQL 7.4«, eine Übersicht über neue Techniken. Dazu gehören Neuerungen in Open SQL sowie datenbanknahe Programmierung.

Teil III: Performantes ABAP

Im vierten Teil befassen wir uns mit dem Thema sicheres ABAP-Programmieren. Zunächst stellen wir Ihnen in **Kapitel 11**, »Stellenwert von Sicherheitsaspekten in ABAP-Programmen«, die Frage, ob Ihre ABAP-Programme verwundbar sind. Neben einer Übersicht über allgemeine Angriffsarten stellen wir allgemeine Lösungsansätze vor und geben Ihnen Empfehlungen, wie Sie Ihre eigenen Programme vor Missbrauch schützen können. In **Kapitel 12**, »Schwachstellen und Schutzmaßnahmen«, stellen wir Ihnen einzelne Sicherheitsrisiken (wie z. B. SQL-Injections, Directory Traversal, Berechtigungsprüfungen, Cross Site Scripting, OS-Command-Injections) zunächst an konkreten Beispielen vor. Danach zeigen wir, welche Ansätze Ihnen helfen, solche Risiken in Ihren ABAP-Programmen zu vermeiden und wie die Korrektur der Beispiele aussehen könnte. In **Kapitel 13**, »Sicherheitsprüfungen für den ABAP-Quelltext durchführen«, stellen wir Ihnen ein SAP-Produkt (SAP NetWeaver Application Server, add-on for code vulnerability analysis) vor, mit dem Sie ABAP-Quelltext auf Schwachstellen hin analysieren können, und geben Ihnen Tipps, wie Sie solche Werkzeuge am effizientesten einsetzen, um Schwachstellen möglichst früh zu finden und diese möglichst schnell und effektiv zu beheben.

Teil IV: Sicheres ABAP

Teil 5: Praxistipps Im letzten Teil wollen wir Ihnen durch Empfehlungen den Einsatz der im Buch behandelten Themen in der täglichen Entwicklungspraxis erleichtern. Dazu gehen wir in **Kapitel 14**, »Qualitätsaspekte in der Projektplanung berücksichtigen«, zunächst auf organisatorische und technische Voraussetzungen für das Umsetzen einer effektiven Qualitätssicherung ein und geben Ihnen für verschiedene Projektarten jeweils eine kurze Checkliste an die Hand. Zum Abschluss des Buches werden wir in **Kapitel 15**, »Empfehlungen für den Einsatz der Techniken und Werkzeuge«, einige Tipps und konkrete Vorgehensweisen besprechen und die wichtigsten Best Practices für »Besseres ABAP« noch einmal kompakt zusammenfassen.

Spezielle Icons Um Sie auf wichtige Informationen hinzuweisen und Ihnen so die Arbeit mit diesem Buch zu erleichtern, verwenden wir im Text die folgenden Icons:

[+] **Tipp**

Kästen mit diesem Icon geben Ihnen Empfehlungen zu Einstellungen oder Tipps aus der Berufspraxis.

[!] **Achtung**

Kästen mit diesem Icon geben Ihnen besonders wichtige Hinweise zur besprochenen Thematik. Außerdem warnen wir Sie hier vor möglichen Fehlerquellen.

[zB] **Beispiel**

Dieses Icon weist Sie auf ausführliche Beispiele hin.

[*] **Weiterführende Informationen**

Mit diesem Icon haben wir zusätzliche, tiefergehende Informationen gekennzeichnet.

Wir hoffen, dass wir Ihnen mit diesem Buch viele wertvolle und direkt umsetzbare Empfehlungen für gute ABAP-Entwicklung an die Hand geben können, und wir wünschen Ihnen viel Spaß bei der Lektüre!

Danksagung

Wir möchten uns bei den folgenden Personen bedanken, die uns durch Diskussionen, Ratschläge und Feedback bei der Erstellung dieses Buches unterstützt haben:

Georg Becker, Immo-Gert Birn, Wulf Bremm, Ulrich Brink, Boris Gebhardt, Patrick Hildenbrand, Holger Janz, Martin Steinberg, Wolf Hagen Thuemmel, Christian Wippermann, Sigrid Wortmann, Mikhail Yakovlev und Klaus Ziegler.

Vielen Dank dafür – ohne eure Hilfe wäre dieses Buch nicht möglich gewesen.

Weiterhin möchten wir uns bei Kerstin Billen für ihre Hilfe bei der Erstellung des Manuskripts und für das Lektorat bedanken.

Ein besonders herzlicher Dank geht auch an unsere Familien für ihr Verständnis für so manche Wochenendarbeit oder Nachtschicht.

**Robert Arlitt, Thorsten Marcus Dunz, Hermann Gahm,
Damir Majer, Eric Westenberger**