

Custom-Code-Migration nach SAP S/4HANA[®]

- › Anpassung Ihrer ABAP-Eigenentwicklungen beim Umstieg auf SAP S/4HANA
- › Optimale Anwendung von ABAP Test Cockpit und Quick Fixes
- › Mit vielen Beispielen und praktischen Empfehlungen

Emily Celen
Lutz Rosenpflanze

Kapitel 2

Das Konvertierungsprojekt im Überblick

In jedem Konvertierungsprojekt gibt es typische wiederkehrende Aufgaben, wenngleich jedes Konvertierungsprojekt unterschiedlich ist. Eine gute Vorbereitung ist ein wesentlicher Erfolgsfaktor.

Viele SAP-Bestandskunden präferieren laut einer IDC-Studie aus dem Juni 2019 (<http://s-prs.de/763207>) eine Systemkonvertierung gegenüber den anderen Transformationsszenarien. Durch eine Systemkonvertierung erhoffen sie sich eine kürzere Projektlaufzeit sowie eine Reduzierung der Aufwände, da bei einem solchen technischen System-Upgrade die Daten und die Konfiguration des Ausgangssystems weitestgehend beibehalten werden können. Jedes Konvertierungsprojekt besteht aus unterschiedlichen Phasen, die wiederum auf eine Vielzahl von einzelnen Aktivitäten heruntergebrochen werden können. Ein Konvertierungsprojekt sollte nicht mit einem einfachen Upgrade-Projekt verwechselt und dementsprechend eingeplant werden. Aufgrund der Simplifizierungsmaßnahmen in SAP S/4HANA sind die für ein Konvertierungsprojekt erforderlichen notwendigen Aktivitäten wesentlich umfangreicher als bei einem typischen Upgrade-Projekt.

In diesem Kapitel geben wir Ihnen einen Überblick über den Ablauf eines Konvertierungsprojekts. Am Beispiel eines groben Projektplans für ein fiktives Unternehmen erläutern wir die Aktivitäten, die für eine SAP-S/4HANA-Systemkonvertierung notwendig sind. Um Ihnen einen besseren Einblick zu geben, welche Änderungen im Kontext der Custom-Code-Migration relevant werden, stellen wir Ihnen in diesem Kontext die zehn wichtigsten Simplification Items vor. Abschließend geben wir Ihnen Tipps zu vorbereitenden Tätigkeiten, die Sie auf Ihrem Transformationspfad nach SAP S/4HANA unterstützen.

2.1 Ablauf eines Konvertierungsprojekts

Bei jeder Systemkonvertierung ist im Vorfeld eine umfassende Analyse des Quellsystems notwendig. In Abbildung 2.1 werden die typischen Phasen eines Konvertierungsprojekts veranschaulicht.

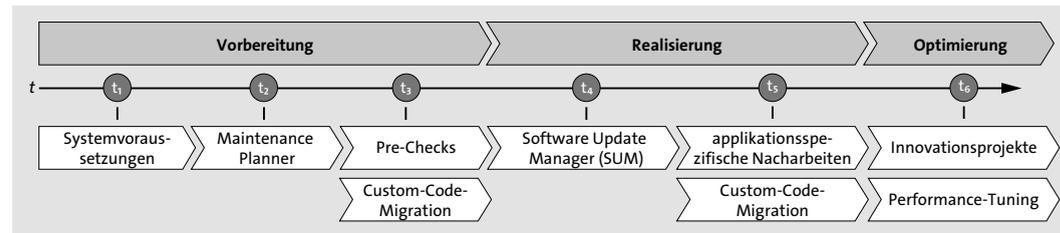


Abbildung 2.1 Typische Phasen eines Konvertierungsprojekts

Vorbereitungsphase In der *Vorbereitungsphase* wird das System auf mögliche Inkompatibilitäten mit SAP S/4HANA überprüft. Die Befunde der Pre-Checks müssen auf ihre Relevanz hin überprüft und bearbeitet bzw. Inkompatibilitäten behoben werden, bevor eine Systemkonvertierung überhaupt möglich ist. SAP bietet verschiedene Tools an, die Sie bei dieser Analyse unterstützen. Über den *SAP Readiness Check for SAP S/4HANA* (siehe Abschnitt 2.4.1) können beispielsweise bestimmte obligatorische Voraussetzungen und Abhängigkeiten für einen Umstieg auf SAP S/4HANA veranschaulicht werden.

Für die Konvertierung eines SAP-ERP-Systems in ein SAP-S/4HANA-System muss das System bereits auf Unicode umgestellt worden sein. Darüber hinaus ist der Umstieg auf SAP S/4HANA nur mit bestimmten Start-Releases möglich (siehe Abschnitt 1.2.2, »Custom-Code-Migration im Rahmen einer Systemkonvertierung«).

Maintenance Planner, SI-Checks und CCC Im nächsten Schritt kann der Maintenance Planner Add-ons und Business Functions auf eine etwaige Inkompatibilität mit SAP S/4HANA hin prüfen und alle notwendigen Softwarepakete für die Aktualisierung auf SAP S/4HANA zusammenstellen. Parallel zu den Simplification Item Checks (SI-Checks) kann bereits die Bearbeitung der Eigenentwicklungen auf Basis des Custom Code Checks (CCC) erfolgen. Viele Befunde, die über den Custom Code Check ermittelt werden, können bereits im SAP-ERP-System bearbeitet werden. Mit SAP ERP 6.0 Enhancement Package (EHP) 8 stellt SAP eine Vielzahl von neuen Klassen und Funktionsbausteinen zur Verfügung, die das Lesen oder Ändern von Simplification Items sowohl in SAP ERP als auch in SAP S/4HANA erlauben.

Software Update Manager In der Phase des Software Update Managers (SUM) findet die eigentliche technische Konvertierung des SAP-ERP-Systems in ein SAP-S/4HANA-System statt. In dieser Phase kann auch die Datenmigration nach SAP HANA durchgeführt werden, sofern noch nicht bereits geschehen.

In der Phase der technischen Konvertierung wird eine *Schatteninstanz* aufgebaut. Dies geschieht durch das Kopieren aller Repository-relevanten Tabellen in ein eigenes Datenbankschema. Ab diesem Zeitpunkt wird die Entwicklungsumgebung durch den SUM gesperrt. In der Schatteninstanz

wird der Abgleich von ABAP-Dictionary-Objekten mit der Transaktion SPDD ausgeführt. Ihre Modifikationen an ABAP-Dictionary-Objekten werden entweder auf den SAP-Standard zurückgesetzt oder manuell mit Ihren Änderungen am ABAP-Dictionary-Objekt abgeglichen, um diese für das SAP-S/4HANA-System zu erhalten.

Anschließend kommt es zu einer *Downtime*, in der eine Anmeldung am System nicht mehr möglich ist. Während dieser Downtime werden unter anderem die Repository-relevanten Tabellen durch die aus der Schatteninstanz ersetzt. Nach der Downtime erfolgen technische Nacharbeiten, und der Abgleich über die Transaktionen SPAU und SPAU_ENH kann begonnen werden. Beim SPAU-Abgleich werden Repository-Objekte entweder auf den SAP-Standard zurückgesetzt oder manuell mit Ihren Modifikationen abgeglichen. Bei Objekten, die mit dem *Modifikationsassistenten* geändert wurden, ist eine automatische Übernahme dann möglich, wenn sich die Modifikationen in der Original- und Kundenversion nicht direkt überschneiden. In SAP S/4HANA bietet die Transaktion SPAU eine komfortable Massenbearbeitung von Objekten an, um die Abgleich-Aufwände zu reduzieren. Für den Abgleich von Erweiterungsimplementierungen wird die Transaktion SPAU_ENH verwendet. Hier können Quelltext-Plug-ins, Enhancements oder BADI-Implementierungen (Business Add-ins) abgeglichen werden.

Parallel zum SPAU-Abgleich kann mit den applikationsspezifischen und technischen Nacharbeiten begonnen werden. Diese Nacharbeiten umfassen beispielsweise die Aktivierung von SAP-Fiori-Apps, die FI-Datenmigration, das Delta-Customizing (z. B. für das Kreditmanagement) oder die Anpassung des Rollenkonzepts. Nach dem der SPAU-Abgleich abgeschlossen ist, folgt eine weitere Phase der Codeanpassungen für SAP S/4HANA.

Änderungszeitpunkt für Befunde

Der Custom Code Check gibt keine Auskunft über den empfohlenen Änderungszeitpunkt für einen Befund. Eine Vielzahl der Anpassungen an den Eigenentwicklungen, die im Rahmen des Custom Code Checks als erforderlich identifiziert werden, können bereits *vor* der technischen Konvertierung mit dem SUM im SAP-ERP-Entwicklungssystem durchgeführt werden. Beispielsweise können viele Befunde im Zusammenhang mit SAP HANA und viele Feldlängenkonflikte bereits im SAP-ERP-System durchgeführt werden. Die betreffenden Anpassungen machen die Eigenentwicklungen meist robuster, verändern aber die Geschäftslogik nicht.

In Ausnahmefällen müssen Änderungen *während* der technischen Konvertierung direkt im jeweiligen Zielsystem ausgeführt werden. Dies betrifft z. B. Append-Erweiterungen der Tabelle MARC (werkspezifischer Materialstamm). Um den zugehörigen Befund zu bearbeiten, ist eine View-Erweite-

ung in SAP S/4HANA nötig, da MARC in SAP S/4HANA keine transparente Tabelle, sondern ein CDS View ist.

Viele weitere Befunde können sinnvollerweise erst *nach* der technischen Konvertierung bearbeitet werden, da erst dann die neuen Strukturen oder Application Programming Interfaces (APIs) zur Verfügung stehen. Direkte Zugriffe auf die Tabelle ACDOCA (das Universal Journal) sind beispielsweise erst nach der Konvertierung möglich, da diese erst in SAP S/4HANA existiert.

Ausgangs- und Zielrelease

Bereits ab SAP ERP 6.0 ohne Enhancement Package ist eine Konvertierung auf SAP S/4HANA möglich. Das Quellsystem kann eine SAP-HANA- oder eine beliebige andere Datenbank (AnyDB) verwenden. Die SAP-HANA-Migration kann direkt im Zuge der technischen Konvertierung durchgeführt werden.

Als Zielrelease kann das aktuellste SAP-S/4HANA-Release festgelegt werden. Denkbar ist es aber auch, zunächst eine Teilkonvertierung, z. B. auf SAP S/4HANA Finance 1605, und anschließend ein System-Upgrade auf das aktuellste SAP-S/4HANA-Release durchzuführen (siehe Abbildung 2.2). Manche Unternehmen präferieren diesen Weg, um die Anzahl der Änderungen bei der Produktivnahme zu reduzieren und damit etwaige Risiken zu minimieren. Diese Entscheidung sollte bewusst getroffen werden und geht immer zulasten der gesamten Projektlaufzeit und den damit verbundenen Projektkosten.

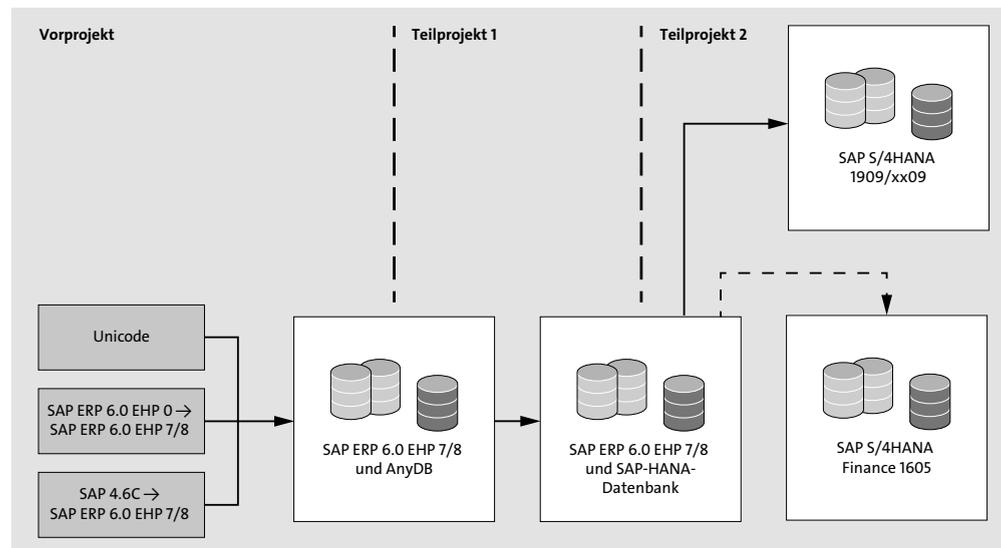


Abbildung 2.2 Konvertierungspfade von SAP ERP nach SAP S/4HANA

Sandbox in Planung berücksichtigen

Wir empfehlen, grundsätzlich bei jeder Systemkonvertierung mindestens die Konvertierung eines Sandbox-Systems in den Projektplan mit aufzuneh-

men und erfolgreich abzuschließen. Auch wenn eine Sandbox-Systemkonvertierung zu einer längeren Projektlaufzeit und höheren Kosten führt, gibt sie die beste Sicherheit, um die spätere Prozessabbildung in SAP S/4HANA zu testen.

Ausnahme in SAP-Einführungsprojekten

Wir empfehlen zwar grundsätzlich mindestens eine Sandbox-Systemkonvertierung, aber auch hier gibt es Ausnahmen, beispielsweise, wenn ein Unternehmen sich gerade mitten in einem Einführungsprojekt für SAP ERP befindet und noch keine produktive Umgebung existiert. Da in diesem Fall der bisherige Arbeitsstand nicht verloren gehen soll, kann eine Systemkonvertierung im Zuge der SAP-Neueinführung durchgeführt werden. In der Regel reicht es in einem solchen Fall aus, ein Backup vom Entwicklungssystem zu erstellen und die Systemkonvertierung anschließend direkt auf Basis des Entwicklungssystems abzuschließen. Da das Entwicklungssystem in der Regel kaum Daten enthält, ist hierdurch ein wesentlich schnellerer Abschluss der Systemkonvertierung möglich. Eine zusätzliche Sandbox-Systemkonvertierung würde den Abschluss des Gesamtprojekts dagegen nur unnötig verzögern.

Wenn zu Beginn eines Konvertierungsprojekts die obligatorischen Voraussetzungen geprüft wurden und mögliche Vorarbeiten im SAP-ERP-System abgeschlossen sind, wird das erste Sandbox-System (SBX1, siehe ❶ in Abbildung 2.3) aus einer aktuellen Systemkopie des produktiven SAP-ERP-Systems (PRD) erstellt. Der Ablauf der Systemkonvertierung wird dann vollständig auf dieser PRD-Kopie durchgespielt.

War die Systemkonvertierung des Sandbox-Systems erfolgreich, kann die Systemkonvertierung des *Entwicklungssystems* (DEV) beginnen ❸. Falls die Systemkonvertierung der Sandbox nicht erfolgreich war, ist der Aufbau eines weiteren Sandbox-Systems (SBX2) notwendig ❷.

Mit wie vielen Sandbox-Systemen man in einem Systemkonvertierungsprojekt planen sollte, ist von unterschiedlichen Faktoren abhängig, wie beispielsweise den vorhandenen Erfahrungen des Projektteams, dem zur Verfügung stehenden Zeitfenster für die Downtime, den zur Verfügung stehenden Ressourcen, der Anzahl und Komplexität der im System abgebildeten Prozesse und Entwicklungen und vielem mehr. Häufig reichen bei mittelständischen Unternehmen ein bis zwei Sandbox-Systemkonvertierungen, bis die SAP-ERP-Systemlandschaft konvertiert werden kann. Letztendlich kann sich aber die Zahl der notwendigen Sandbox-Systeme auch ungeplant erhöhen, z. B. wenn aufgrund von Ressourcenumstellungen, schlechter Dokumentation, stetig wechselnder Umfänge, inakzeptabler Downtime-Zeiten oder Ähnli-



Anzahl der Sandbox-Systeme

chem der Ablauf der Konvertierung immer wieder neu durchexerziert werden muss (siehe auch Abschnitt 9.1.1, »Organisatorische Herausforderungen und Fallstricke«).

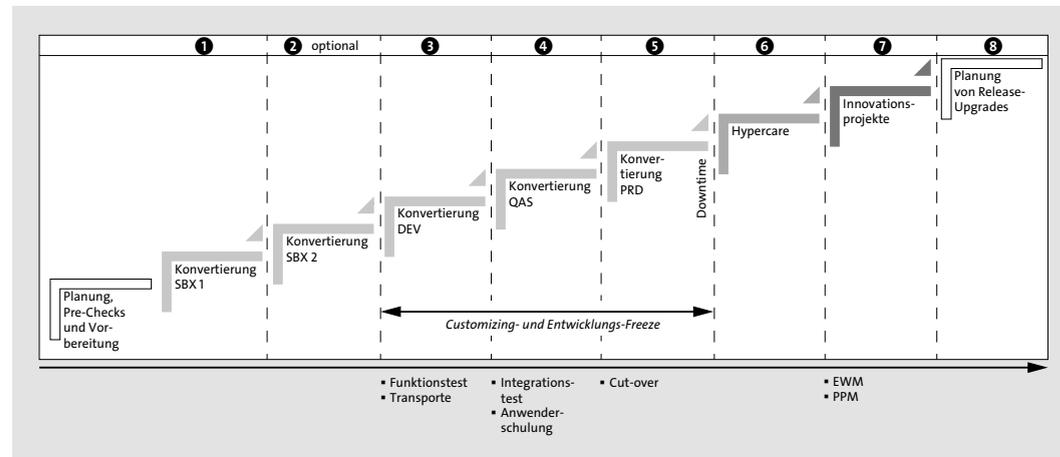


Abbildung 2.3 Ablauf einer Systemkonvertierung mit Sandbox-Systemen

System-Freeze Sobald die Systemkonvertierung des Entwicklungssystems begonnen wird, gilt ein allgemeiner *System-Freeze*, das heißt, Customizing- und Workbench-Änderungen sind nicht mehr zulässig. In Einzelfällen können dringende Transporte in das produktive SAP-ERP-System übernommen werden, wenn gewährleistet ist, dass die jeweilige Änderung keine negativen Seiteneffekte hat. Wenn beispielsweise die ursprüngliche Version des betroffenen Repository-Objekts in Workbench-Transporten oder die Struktur von Customizing-Tabellen in Customizing-Transporten in SAP ERP und in SAP S/4HANA nicht identisch sind, kann es zu Importabbrüchen oder Laufzeitfehlern kommen.

Konvertierung der SAP-ERP-Systemlandschaft Nachdem das Entwicklungssystem konvertiert und der Funktionstest erfolgreich abgeschlossen wurde, folgt analog die Systemkonvertierung des *Qualitätssicherungssystems* (QAS 4) mit einem umfangreichen Integrationstest der Prozesse und angebundenen Schnittstellen.

Anschließend wird das Produktivsystems (PRD 5) nach SAP S/4HANA konvertiert. Alle notwendigen Tätigkeiten für die Vor- bzw. Nachbearbeitung werden über einen Cut-over-Plan gesteuert. Wie bei jedem anderen größeren Projekt sollte nach dem *Go-live* des Produktivsystems mit einer *Hypercare-Phase* 6 von mindestens zwei Wochen geplant werden. In dieser Phase muss das Projektteam bei etwaigen Problemen schnell aktiv werden können. Zuletzt folgen in der längerfristigen Planung weitere Innovationsprojekte 7, wie beispielsweise die Einführung von SAP Extended Warehouse Management (SAP EWM) oder SAP Portfolio and Project Management (SAP PPM), und weitere Updates bzw. Upgrades 8 des SAP-S/4HANA-Systems.

2.2 Beispiel für einen groben Projektplan

Ein erster grober Projektplan kann bereits sehr früh zu Projektbeginn erstellt werden, ohne dass die Details der Prozessabbildung in SAP S/4HANA bereits bekannt sind. Der Projektplan umfasst die einzelnen Projektphasen mit ersten Indikatoren für die Laufzeit. Abhängig von der Anzahl der Ergebnisse aus dem SAP Readiness Check, dem Custom Code Check und dem SI-Check können so bereits sehr früh die ungefähren Aufwände für das Projekt eingeplant werden. Darüber hinaus kann eine erste Schätzung erfolgen, mit wie vielen Sandbox-Systemen man für das Transformationsprojekt planen sollte.

In der Praxis tendieren mittelständische Unternehmen häufig zu mehreren, kleineren Vorprojekten, in denen technische oder fachliche bzw. prozessspezifische Änderungen bereits in SAP ERP produktiv gesetzt werden, anstatt mit einem großen *Big Bang* alle technischen und fachlichen Änderungen auf einmal einzuführen. Durch die Splittung des Transformationsprojekts in einfacher kalkulierbare Vorprojekte können etwaige Komplikationen bei der Produktivsetzung besser überwacht und gegebenenfalls bearbeitet werden.

In der Regel erhöhen sich durch jedes Vorprojekt aber auch die *Projektlaufzeit* und damit verbunden die *Projektgesamtkosten*. Die Synergien eines Gesamtprojekts können nicht mehr effizient genutzt werden. Beispielsweise durchläuft jedes umfangreichere Projekt in der Regel einen Integrationstest vor dem *Go-live*. Bei der Projektplanung mit Vorprojekten müssen für jedes Vorprojekt Spezifizierungs-, Implementierungs-, Tests- und *Go-live*- bzw. Hypercare-Phasen in den Gesamtprojektplan mit aufgenommen werden.

In der Praxis werden häufig die SAP-HANA-Migration oder die Aktivierung des Geschäftspartners in SAP ERP als Vorprojekte durchgeführt, bevor das eigentliche Transformationsprojekt gestartet wird. Ob und welche Vorprojekte im Gesamtprojekt eingeplant werden, ist von vielen Faktoren abhängig, wie Ressourcen, Verfügbarkeit, Ausgangssystem(e), Zielsystem(e), produktiv verwendete Komponenten oder dem Budget. Eine generelle Empfehlung ist daher nicht möglich.

Reduzierung von Vorprojekte auf ein sinnvolles Maß

Grundsätzlich sollte die Anzahl der Vorprojekte auf ein sinnvolles Maß reduziert bzw. sollten solche Projekte möglichst parallelisiert werden, um Synergien im Gesamtprojekt effektiv nutzen zu können.

Abbildung 2.4 zeigt einen beispielhaften groben Projektplan des fiktiven Unternehmens *Sonnenschein AG*.

Big Bang vs. Vorprojekte



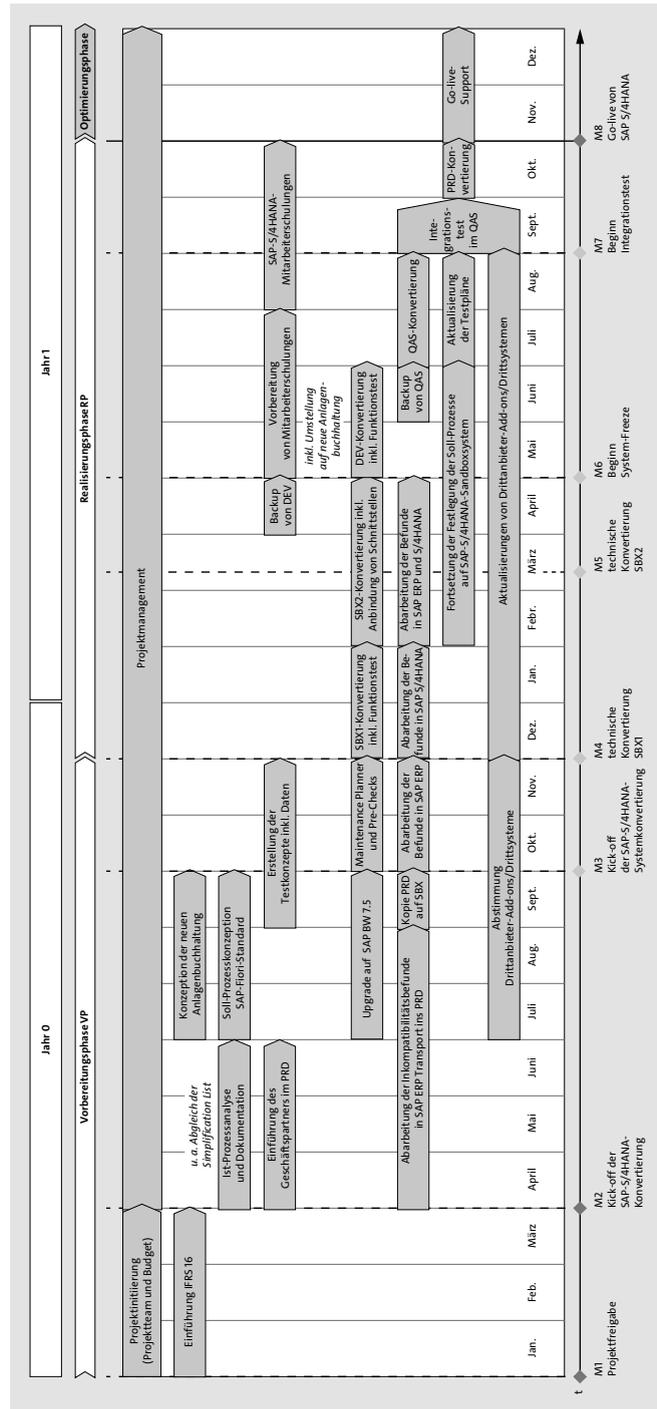


Abbildung 2.4 Beispiel eines groben Projektplans für ein mittelständisches Unternehmen

Bei der Sonnenschein AG handelt es sich um ein mittelständisches, international tätiges Unternehmen aus der Fertigungsindustrie, das SAP ERP 6.0 EHP 8 mit einer SAP-HANA-Datenbank im Einsatz hat. Die Sonnenschein AG nutzt das SAP-ERP-System bereits seit 15 Jahren und hat mithilfe der zwei bis drei SAP-Entwickler in der eigenen IT-Abteilung stetig Erweiterungen an diesem System durchgeführt.

Die Sonnenschein AG nutzt die Systemkonvertierung nach SAP S/4HANA als Chance, um die eigenen Prozesse zu verbessern und teilweise auch wieder zurück in den SAP-Standard zu überführen. Da mehrere andere Projekte parallel laufen, befürchtet sie jedoch, dass es bereits zu Projektbeginn zu Ressourcenengpässen hinsichtlich der gefragten Know-how-Träger kommt. Da das Unternehmen rund um die Uhr produziert, würden ein längerer Systemausfall oder Komplikationen durch Änderungen zu erheblichen Problemen führen. Aus diesem Grund werden verschiedene Vorprojekte in den Projektplan aufgenommen und die einzelnen Phasen jeweils mit einem zeitlichen Puffer versehen. Der Projektplan sieht für die gesamte Systemkonvertierung einen Zeitraum von ca. zwei Jahren vor.

Wegen gesetzlicher Änderungen muss die Sonnenschein AG zunächst die Bilanzierung nach dem *International Financial Reporting Standard 16* (IFRS 16) einführen. Dieses Vorprojekt muss abgeschlossen sein, bevor das eigentliche SAP-S/4HANA-Konvertierungsprojekt begonnen werden kann.

Nach Abschluss der IFRS-16-Einführung startet die Sonnenschein AG eine Analyse und Dokumentation der aktuellen Prozesse. Erst wenn dieser Ist-Stand bekannt ist, können darauf aufbauend die unternehmensspezifischen Soll-Prozesse in SAP S/4HANA definiert werden. Parallel zu dieser Bestandsaufnahme wird bereits der Geschäftspartner in SAP ERP aktiviert und im Produktivsystem eingeführt.

Die Bearbeitung von Befunden aus dem Custom Code Check ist von den zuvor genannten Tätigkeiten weitestgehend unabhängig. Die Sonnenschein AG kann über den Einsatz der App *Custom Code Migration* nachvollziehen, dass viele Entwicklungen nicht mehr benötigt werden. In einer ersten Phase werden deshalb nicht mehr benötigte Entwicklungen massenhaft stillgelegt. Anschließend möchte die Sonnenschein AG mit der Anpassung der verbleibenden Entwicklungen für SAP S/4HANA beginnen. Die internen Entwickler wissen, dass sie in der aktuellen Projektphase nur Codeänderungen durchführen dürfen, die sowohl mit SAP ERP als auch mit SAP S/4HANA kompatibel sind. Diese Codeänderungen werden in Paketen durchgeführt, getestet und in das Produktivsystem übernommen.

Anschließend folgt die erste Sandbox-Systemkonvertierung. Jeder Fehler, der bei dieser Konvertierung auffällt, und jede daraufhin vorgenommene

Ausgangslage der Sonnenschein AG

Vorprojekt IFRS 16

Ist- und Soll-Prozesse

Beginn der Custom-Code-Anpassungen

Dokumentation der Änderungen

Änderung wird von der Sonnenschein AG dokumentiert oder direkt im produktiven SAP-ERP-System behoben, um sie anschließend bei der zweiten Sandbox-Systemkonvertierung zu validieren.

Da die zweite Sandbox-Systemkonvertierung ein zufriedenstellendes Ergebnis liefert, folgt anschließend die Systemkonvertierung des Entwicklungssystems. Auf dem Entwicklungssystem wird ein Funktionstest durchgeführt. Danach wird das Qualitätssicherungssystem konvertiert. Im Qualitätssicherungssystem werden die Schnittstellen angebunden, und der Integrationstest wird durchgeführt. Abschließend folgen die Systemkonvertierung des Produktivsystems und eine zweimonatige Hypercare-Phase.

2.3 Welche Änderungen am Anwendungscode sind erforderlich?

In diesem Abschnitt geben wir Ihnen einen Einblick in die notwendigen Änderungen von Eigenentwicklungen im Zuge einer SAP-S/4HANA-Transformation. Einen ersten Überblick über den Umfang der technischen Änderungen erhalten Sie über die Vereinfachungsdatenbank (*Simplification Database*), die wir in Abschnitt 2.3.1 vorstellen. Die Vereinfachungsdatenbank dient als Basis für den Custom Code Check und beinhaltet alle technischen Änderungen in SAP S/4HANA.

Um den Umstellungsaufwand für die Eigenentwicklungen zu reduzieren, stellt SAP spezielle Kompatibilitätsobjekte zur Verfügung, die wir in Abschnitt 2.3.2, »Kompatibilitäts-Views«, beschreiben. In manchen Fällen kann eine Änderung der Eigenentwicklungen umgangen werden, indem man auf diese speziellen Objekte zugreift, die trotz der Änderung des Datenmodells auch mit SAP S/4HANA weiterhin kompatibel sind.

Abschließend geben wir in Abschnitt 2.3.3, »Die Top 10 der Simplification Items«, einige Beispiele für mögliche Anpassungen von Eigenentwicklungen. Alle möglichen Änderungen, die im Zuge einer SAP-S/4HANA-Transformation erforderlich werden können, wären zu divers, um sie hier zu beschreiben. Daher konzentrieren wir uns hier auf die Behandlung der am häufigsten auftretenden Befunde.

2.3.1 Die Vereinfachungsdatenbank

Die Vereinfachungsdatenbank fasst alle technischen Änderungen von SAP S/4HANA im Vergleich zu SAP ERP zusammen. Jede einzelne aufgeführte Änderung wird als Simplification Item bezeichnet. Die Simplification Items werden in der Regel über SAP-Hinweise näher beschrieben. Für jedes neue

Technische Basis
für den Custom
Code Check

Major Release von SAP S/4HANA wird eine neue Simplification List von SAP veröffentlicht, mit einem stetig wachsendem Inhalt und neuen Simplification Items. Die Simplification List ist eine textuelle Beschreibung der Änderungen in SAP S/4HANA im Vergleich zu SAP ERP. Im Gegensatz dazu werden in der Vereinfachungsdatenbank die technischen Objekte (Transaktionen, Reports, Funktionen und vieles mehr) zusammengefasst, die von den Änderungen in SAP S/4HANA betroffen sind.

Die Vereinfachungsdatenbank bildet die Grundlage für den Custom Code Check. Sie kann direkt aus dem *SAP Software Download Center* (<http://s-prs.de/v763208>) heruntergeladen und anschließend über die Transaktion SYCM im Prüfsystem importiert werden (Details hierzu lesen Sie in Abschnitt 4.2.1, »Vorbereitungen zur Nutzung des ABAP Test Cockpits«). In Abbildung 2.5 wird ein Auszug der Vereinfachungsdatenbank über die Transaktion SYCM angezeigt.

Download

SAP-Objektname	Anwendungskomponente	Vereinfachungskategorie	SAP-Hinweisnummer	SAP-Hinweistitel	Prüfkategorie
FUNC CS_HBM_MBOB_EXISTENCE_CHECK	CA-FILE-MAT	Syntaktisch nicht kompatible Änderung d. vorhandenen Funktion	2438131	Feldlängenerweiterung für Materialnummer: Quelltextanpassungen für Verwendungen	
FUNC BAPI_NETWORK_GETDETAIL	CA-FILE-MAT				
FUNC BAPI_KANBANCC_GETLIST_ALL	CA-FILE-MAT				
FUNC CAVC_C_CHECK_INSTANTIATION	CA-FILE-MAT				
DOMA OBJKN	PM-WOC-MO		2670006	Feldlängenerweiterung für Objektlistennummer - allgemeine technische Hintergrund	1
CLAS CL_CS_BOM_PRODVER_MIGRATION	LO-MD-BOM		2215123	Fertigungsstücklistenauflösung wird über Fertigungsversionen empfohlen	
DOMA CS_BOM_UPDATED_FIELD	LO-MD-BOM				
DOCU INACS_BOM_HANAD15	LO-MD-BOM				
DOCU RECS_BOM_PRODVER_MIGRATION	LO-MD-BOM				
MESS CS_BOM_HANAD15	LO-MD-BOM				
REPT CL_CS_BOM_PRODVER_MIGRATION=	LO-MD-BOM				
FUNC CS_BOM_STKO_UPDATE	LO-MD-BOM				
CLSD CL_CS_BOM_PRODVER_MIGRATION	LO-MD-BOM				
PROG CS_BOM_PRODVER_MIGRATION_TOP	LO-MD-BOM				
FUNC CS_BOM_STKO_MASS_VALIDITY	LO-MD-BOM				
DTCL CS_PRODVER_NEW	LO-MD-BOM				
TABL BOM_TY_PRODVER_MATL_ENTRIES	LO-MD-BOM				
PROG CS_BOM_PRODVER_MIGRATION	LO-MD-BOM				
FUNC CS_BOM_STKO_VALIDITY	LO-MD-BOM				
FLGR CS_BOM	LO-MD-BOM				

Abbildung 2.5 Inhalt der Vereinfachungsdatenbank

Weitere Informationen zum Download der Vereinfachungsdatenbank

SAP-Hinweis 2241080 enthält detaillierte Informationen zum Download der Vereinfachungsdatenbank.



Zum Zeitpunkt der Drucklegung dieses Buches umfasst die Vereinfachungsdatenbank ca. 475.200 Objekte. Das Volumen der Vereinfachungsdatenbank ist ein Indikator dafür, wie umfangreich die Erweiterungen und Vereinfachungen in SAP S/4HANA im Vergleich zu SAP ERP sind. Der Einsatz der Vereinfachungsdatenbank ist für den Custom Code Check obligatorisch, da es aufgrund der Vielzahl an Änderungen nicht möglich wäre, einen manuellen Abgleich auszuführen.

2.3.2 Kompatibilitäts-Views

Gleichbleibender Datenzugriff

Im Zuge der technischen Umstrukturierung des Datenmodells für die Einführung von SAP Simple Finance wurden viele Summen- und Anwendungstextabellen eliminiert. Aus Kompatibilitätsgründen wurden sogenannte *Kompatibilitäts-Views* eingeführt, mit denen ein ähnlicher Zugriff auf die Daten wie vor der Umstrukturierung möglich ist. Die Kompatibilitäts-Views sind CDS Views, die die Daten aus den neuen Datenstrukturen im Format der alten Datenbanktabellen abbilden. Sie haben teilweise denselben technischen Namen wie die ursprüngliche Datenbanktabelle.



Zwei Versionen von Kompatibilitäts-Views

Im Zuge der Entwicklung von SAP Simple Finance hat SAP zwei unterschiedliche technische Varianten verwendet, um die Kompatibilitäts-Views zu implementieren. Die Kompatibilitäts-Views der *Version 1 (V1)* wurden mit dem ersten Release des SAP-Simple-Finance-Add-ons eingeführt. Sie tragen immer den gleichen Namen wie die ursprünglichen Datenbanktabellen. Sie werden bei einer Systemkonvertierung über den SUM automatisch generiert. Der alte Datenbestand wird in Backup-Tabellen gesichert. Die Namen dieser Backup-Tabellen beginnen jeweils mit dem Namen der Originaltabelle, gefolgt von dem Suffix `_BCK`. Für die Tabelle BSAD wird beispielsweise die Backup-Tabelle BSAD_BCK angelegt.

Mit SAP Simple Finance 2.0 bzw. SAP S/4HANA 1511 wurde die *Version 2 (V2)* der Kompatibilitäts-Views eingeführt. Diese Views unterscheiden sich in folgenden Aspekten von den Views der V1:

- Die Kompatibilitäts-Views V2 haben nicht den gleichen Namen wie die ursprünglichen Datenbanktabellen.
- Die Datenbankschnittstelle leitet automatisch alle SELECT-Anfragen an die ursprünglichen Datenbanktabellen auf den entsprechenden Kompatibilitäts-View um. Die Kompatibilitäts-Views V2 werden daher auch als *Redirect-Views* bezeichnet.

Die Ermittlung eines Kompatibilitäts-Views V1 ist etwas einfacher bzw. offensichtlicher als die Ermittlung eines Views der V2-Variante. Dafür kommt es bei speziellen, inkorrekten Zugriffen auf Kompatibilitäts-Views V2 nicht zu Laufzeitfehlern, wie das bei der V1 der Fall war.

Originaltabellen zu Kompatibilitäts-Views V1

In Tabelle 2.1 sind die Originaltabellen aufgeführt, die im Zuge der Entwicklung von SAP Simple Finance durch gleichnamige Kompatibilitäts-Views V1 ersetzt wurden.

Original-tabelle	Beschreibung	CDS View
BSAD	Buchhaltung: Sekundärindex für Debitoren (ausgebliebene Posten)	BSAD_DDL
BSAK	Buchhaltung: Sekundärindex für Kreditoren (ausgeglichene Posten)	BSAK_DDL
BSAS	Buchhaltung: Sekundärindex für Sachkonten (ausgeglichene Posten)	BSAS_DDL
BSID	Buchhaltung: Sekundärindex für Debitoren	BSID_DDL
BSIK	Buchhaltung: Sekundärindex für Kreditoren	BSIK_DDL
BSIS	Buchhaltung: Sekundärindex für Sachkonten	BSIS_DDL
FAGLBSAS	Buchhaltung: Sekundärindex für Sachkonten (ausgeglichene Posten)	FAGLBSAS_DDL
FAGLBSIS	Buchhaltung: Sekundärindex für Sachkonten	FAGLBSIS_DDL
GLTO	Sachkontenstamm Verkehrszahlen	GLTO_DDL
KNC1	Kundenstamm Verkehrszahlen	KNC1_DDL
KNC3	Kundenstamm Verkehrszahlen Sonderhauptbuchvorgänge	KNC3_DDL
COSP	CO-Objekt: Summen Kosten – externe Buchungen	V_COSP_DDL
COSS	CO-Objekt: Summen Kosten – interne Buchungen	V_COSS_DDL
FAGLFLEXT	Hauptbuch: Summen	V_FAGLFLEXT_DDL

Tabelle 2.1 Kompatibilitäts-Views V1 und zugehörige Originaltabellen

Die ergänzenden Kompatibilitäts-Views V2 und deren Originaltabellen sind in Tabelle 2.2 aufgeführt.

Originaltabellen zu Kompatibilitäts-Views V2

Original-tabelle	Beschreibung	CDS View
ANEA	Anlagen-Einzelposten anteilige Werte	FAAV_ANEA
ANEK	Belegkopf Anlagenbuchung	FAAV_ANEK

Tabelle 2.2 Kompatibilitäts-Views V2 und zugehörige Originaltabellen

Original-tabelle	Beschreibung	CDS View
ANEP	Anlagen-Einzelposten	FAAV_ANEP
ANLC	Anlagen-Wertfelder	FAAV_ANLC
ANLP	Anlagen-Periodenwerte	FAAV_ANLP
BSIM	Sekundärindex Belege zum Material	V_BSIM
CKMI1	Index für Rechnungswesenbelege zum Material	V_CKMI1
COEP	CO-Objekt: Einzelposten periodenbezogen	V_COEP
FAGLFLEXA	Hauptbuch: Ist-Einzelposten	FGLV_FAGLFLEXA
MLCD	Material-Ledger: Verdichtungssätze (aus Belegen)	V_MLCD
MLCR	Material-Ledger-Beleg: Währungen und Werte	V_MLCR
MLHD	Material-Ledger-Beleg: Kopf	V_MLHD
MLIT	Material-Ledger-Beleg: Positionen	V_MLIT
MLPP	Material-Ledger-Beleg: Buchungsperioden und Mengen	V_MLPP
T012K	Konten bei Hausbanken	V_T012K_BAM
T012T	Bezeichnung der Konten bei Hausbanken	V_T012T_BAM
FMGLFLEXA	Hauptbuch: Ist-Einzelposten	FGLV_FMGLFLEXA
FMGLFLEXT	Hauptbuch Public Sector: Summen	FGLV_FMGLFLEXT
PSGLFLEXA	Hauptbuch: Ist-Einzelposten	FGLV_PSGLFLEXA
PSGLFLEXT	Hauptbuch Public Sector: Summen	FGLV_PSGLFLEXT
JVGLFLEXA	General Ledger: Actual Line Items	FGLV_JVGLFLEXA
JVGLFLEXT	General Ledger: Totals	FGLV_JVGLFLEXT

Tabelle 2.2 Kompatibilitäts-Views V2 und zugehörige Originaltabellen (Forts.)

Darstellung des Kompatibilitäts-Views V1

Ruft man in Transaktion SE11 eine der Datenbanktabellen auf, für die es einen Kompatibilitäts-Views V1 gibt, wird der Name des CDS Views, über den die Datenermittlung erfolgt, im Feld **DDL Source** angezeigt, wie in Abbildung 2.6 dargestellt.

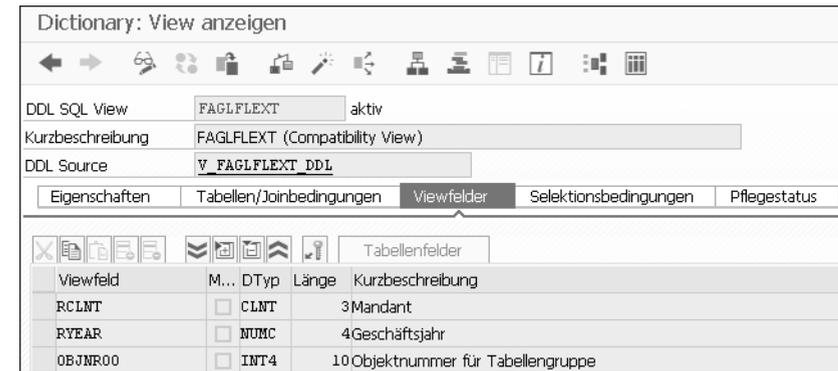


Abbildung 2.6 Aufruf des Kompatibilitäts-Views V2 zur Tabellenstruktur FAGLFLEX

Bei einem Kompatibilitäts-View V2 ist das anders. Wie am Beispiel der Tabelle FAGLFLEXA in Abbildung 2.7 zu erkennen, suggeriert die Anzeige in Transaktion SE11, dass es sich bei dem aufgerufenen Objekt um eine (unveränderte) transparente Tabelle handelt.

Darstellung des Kompatibilitäts-Views V2

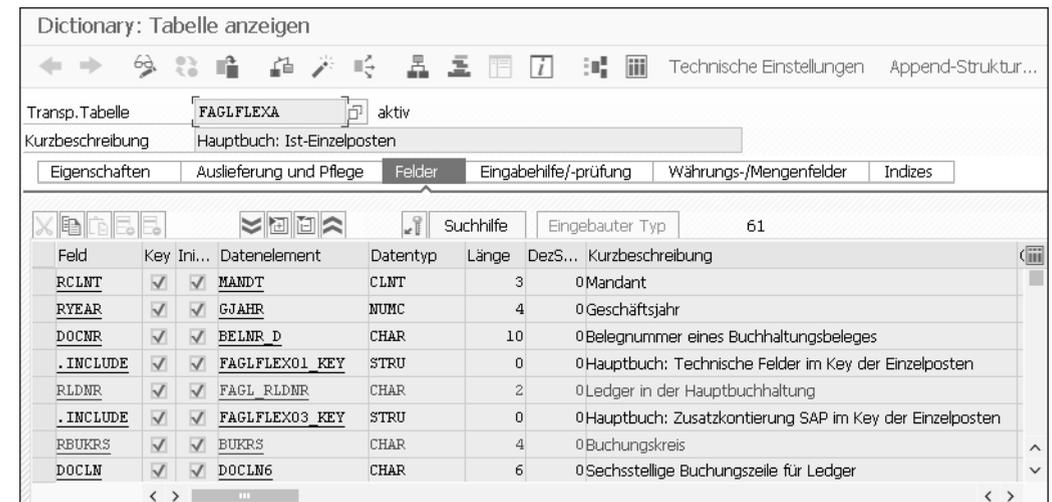


Abbildung 2.7 Aufruf des Kompatibilitäts-Views V2 zur Tabellenstruktur FAGLFLEXA

Bei Kompatibilitäts-Views V2 wird der CDS View nicht angezeigt. Die Darstellung des Datenbanktabelleninhalts ist über die Transaktion SE16N wie gewohnt möglich. Auch können Sie innerhalb einer Eigenentwicklung in der Regel wie gewohnt über eine **SELECT**-Anweisung auf das Datenbankobjekt zugreifen. Das Objekt verhält sich im System scheinbar exakt so, als würde die alte transparente Datenbanktabelle FAGLFLEXA noch existieren.

CDS View anzeigen Die »technische Wahrheit« versteckt sich in der Implementierung des CDS Views, der für die neue Datenermittlung verantwortlich ist. Den Namen dieses CDS Views können Sie im SAP GUI auf verschiedenen Wegen ermitteln:

■ **Vertreterobjekt zur Tabelle anzeigen**

In Transaktion SE11 folgen Sie dem Menüpfad **Zusätze • Vertreterobjekt**, um sich den Namen des Kompatibilitäts-Views anzeigen zu lassen. Der Kompatibilitäts-View wird im sich daraufhin öffnenden Pop-up-Fenster im Feld **Vertreterobjekt** angezeigt (siehe Abbildung 2.8).

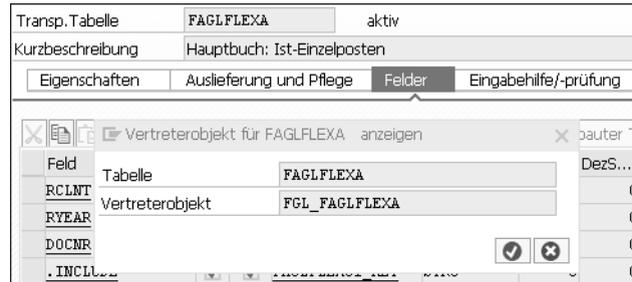


Abbildung 2.8 Anzeige des Kompatibilitäts-Views zu einer Tabelle in Transaktion SE11

■ **Proxy-Objekt zur Tabelle anzeigen**

In Transaktion SE16N oder SE16H geben Sie den Namen der Datenbanktabelle im Feld **Tabelle** ein und drücken die **↵**-Taste. Der Name des Kompatibilitäts-Views wird daraufhin im Feld **Proxy-Objekt** angezeigt (siehe Abbildung 2.9).

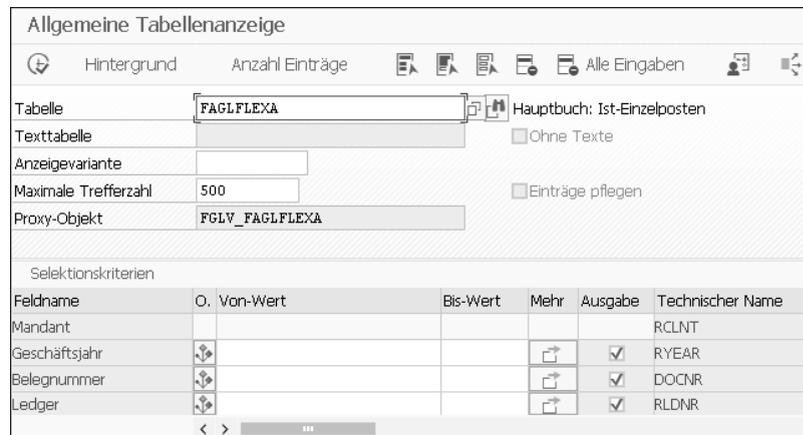


Abbildung 2.9 Anzeige des Kompatibilitäts-Views zu einer Tabelle in Transaktion SE16N

■ **SQL-Trace auswerten**

Über die Transaktion ST05 (*SQL-Trace*) können Datenbankzugriffe protokolliert und die Details zu den Zugriffen angezeigt werden.

Zunächst aktivieren Sie den SQL-Trace über den Button **Trace einschalten**. Dann können Sie beispielsweise über die Transaktion SE16N exemplarisch auf Datenbankinhalte der Tabelle FAGLFLEXA zugreifen. Anschließend deaktivieren Sie den Trace über den Button **Trace ausschalten**.

SQL-Trace auswerten

Über den Button **Trace anzeigen** können Sie nun die Ergebnisse des Trace auswerten. In der Ergebnisliste erscheinen alle Datenbanktabellen, auf die zugegriffen wurde, während der Trace aktiviert war. In der Ergebnisliste wird in der Spalte **Objektname** der Kompatibilitäts-View zur jeweiligen Tabelle angezeigt (siehe Abbildung 2.10).

Startzeit	Dauer	#Sätze	Programmname	Objektname	Anweisung
20:04:45.817	501	1	/1BCDWB/DBFAGLFLEXA	DDNTT	SELECT WHERE TABNAME = 'FAGLFLEX03_DATA'
20:04:45.818	5.345	1	/1BCDWB/DBFAGLFLEXA	DDNTF	SELECT WHERE TABNAME = 'FAGLFLEX03_DATA' ORDER BY TABNAME, BLOCKNR
20:04:45.823	574	1	/1BCDWB/DBFAGLFLEXA	DDNTT	SELECT WHERE TABNAME = 'FAGLFLEX06_INC'
20:04:45.824	1.158	1	/1BCDWB/DBFAGLFLEXA	DDNTF	SELECT WHERE TABNAME = 'FAGLFLEX06_INC' ORDER BY TABNAME, BLOCKNR
20:04:45.826	947	1	/1BCDWB/DBFAGLFLEXA	DDNTT	SELECT WHERE TABNAME = 'FAGLFLEX_SI_VALUE_DATA'
20:04:45.827	1.107	1	/1BCDWB/DBFAGLFLEXA	DDNTF	SELECT WHERE TABNAME = 'FAGLFLEX_SI_VALUE_DATA' ORDER BY TABNAME, BLOCKNR
20:04:45.829	542	1	/1BCDWB/DBFAGLFLEXA	DDNTT	SELECT WHERE TABNAME = 'FAGLFLEX_SI_FIX_INC'
	14.379	1	/1BCDWB/DBFAGLFLEXA	DDNTF	SELECT WHERE TABNAME = 'FAGLFLEX_SI_FIX_INC' ORDER BY TABNAME, BLOCKNR
20:04:45.876	77.394	200	/1BCDWB/DBFAGLFLEXA	FGLV_FAGLFLEXA	SELECT <FDA READ> WHERE 'RCLNT' = '900' LIMIT 200
20:04:45.937	18.376	3	SAPLSETB	D0205	SELECT <FDA READ> WHERE 'PROG' = '/1BCDWB/DBFAGLFLEXA'
20:04:45.945	958	1	SAPLSETB	D347T	SELECT WHERE 'PROGRAMME' = 'SAPLSETB' AND 'SPRSL' = 'D' AND 'OBJ_CODE' = 'TA3'
20:04:45.972	12.031	1	CL_ABAP_LIST_PARSER=====CP	TRDIR	SELECT WHERE 'NAME' = '/1BCDWB/DBFAGLFLEXA' LIMIT 1
20:04:45.980	790	1	CL_DSH_SCREEN_ASSIGNMENT=====CP	D0205	SELECT WHERE 'PROG' = 'SAPMSSYO' AND 'DNUM' = '0120' LIMIT 1
20:04:45.981	1.030	1	CL_DSH_SCREEN_ASSIGNMENT=====CP	D0205	SELECT WHERE 'PROG' = 'SAPMSSYO' AND 'DNUM' = '0120'
20:04:45.983	1.428	1	CL_DSH_SCREEN_ASSIGNMENT=====CP	DYNPSOURCE	SELECT WHERE PROGRAMME = 'SAPMSSYO' AND DYNPNUMBER = '0120' AND R3STATE = 'A'
20:04:45.985	1.518	1	CL_DSH_SCREEN_ASSIGNMENT=====CP	DYNPSOURCE	SELECT WHERE PROGRAMME = 'SAPMSSYO' AND DYNPNUMBER = '0120' AND R3STATE = 'A'
20:04:45.987	1.016	1	CL_DSH_SCREEN_ASSIGNMENT=====CP	DYNPSOURCE	SELECT WHERE PROGRAMME = 'SAPMSSYO' AND DYNPNUMBER = '0120' AND R3STATE = 'A'

Abbildung 2.10 Anzeige des Kompatibilitäts-Views in Transaktion ST05 (Hauptsätze zum SQL-Trace)

Wenn Sie eine Zeile markieren, z. B. die für einen Zugriff auf das Objekt FGLV_FAGLFLEXA, und auf den Button **Anweisungen anzeigen** der ALV-Anzeige klicken, navigieren Sie zur Detailanzeige der verwendeten SQL-Anweisung. Hier werden der Name der ursprünglichen Tabelle (**Redirected table**) und der des Kompatibilitäts-Views, auf den der Zugriff umgeleitet wurde, im Bereich der FROM-Anweisung angezeigt (siehe Abbildung 2.11). Jeder Zugriff auf die Tabelle FAGLFLEXA wird automatisch über die Datenbankschnittstelle an den CDS View FGLV_FAGLFLEXA weitergeleitet.

Über die *ABAP Development Tools* (ADT) in Eclipse können Sie sich die Tabellendefinition in CDS anzeigen lassen (siehe Abbildung 2.12). Die Tabellendefinition beginnt klassisch mit der Anweisung `define table`. Die Annotation `@AbapCatalog.replacementObject` definiert den Kompatibilitäts-View und damit die eigentliche Datenquelle. Für die Tabelle FAGLFLEXA werden die Daten aus dem CDS View FGLV_FAGLFLEXA ermittelt.

Anzeige der Tabellendefinition

SQL Trace-Satz	
Details zum ausgewählten SQL-Trace-Satz	
<pre> SELECT /* FDA READ */ "RCLNT" , "RYEAR" , "DOCNR" , "RLDNR" , "RBUKRS" , "DOCLN" , "ACTIV" , "RMVCT" , "RTCUR" , "RUNIT" , "AWTYP" , "RRCTY" , "RVERS" , "LOGSYS" , "RACCT" , "COST_ELEM" , "RCNTR" , "PRCTR" , "RFAREA" , "RBUA" , "KOKRS" , "SEGMENT" , "SCNTR" , "PPCTR" , "SFAREA" , "SBUA" , "RASSC" , "PSEGMENT" , "TSL" , "HSL" , "KSL" , "OSL" , "MSL" , "WSL" , "DRCRK" , "POPER" , "RMCUR" , "GJAHR" , "BUDAT" , "BELNR" , "BUZEI" , "BSCHL" , "BSTAT" , "LINETYPE" , "XSPLITMOD" , "USNAM" , "TIMESTAMP" , "DATAAGING" FROM /* Redirected table: FAGLFLEXA */ "FGLV_FAGLFLEXA" "FAGLFLEXA" WHERE "RCLNT" = 2 LIMIT 2 WITH RANGE_RESTRICTION('CURRENT') </pre>	
Variablen	
A0(CH,3)	= '900'
A1(I4,3)	= 500

Abbildung 2.11 Anzeige des Kompatibilitäts-Views in Transaktion ST05 (SQL-Trace-Satz)

[SAE] FAGLFLEXA	
1	@EndUserText.label : 'General Ledger: Actual Line Items'
2	@AbapCatalog.enhancementCategory : #EXTENSIBLE_CHARACTER_NUMERIC
3	@AbapCatalog.tableCategory : #TRANSPARENT
4	@AbapCatalog.deliveryClass : #A
5	@AbapCatalog.dataMaintenance : #LIMITED
6	@AbapCatalog.replacementObject : 'fgl_faglflexa'
7	define table faglflexa {
8	@AbapCatalog.foreignKey.keyType : #KEY
9	@AbapCatalog.foreignKey.screenCheck : true
10	key rclnt : mandt not null
11	with foreign key [0..*1] t000
12	where mandt = faglflexa.rclnt;
13	key ryear : gjahr not null;
14	key docnr : belnr_d not null;
15	key include faglflex01_key not null;
16	key include faglflex03_key not null;
17	key docln : docln6 not null;
18	include faglflex00_inc not null
19	extend rtcnr :
20	remove foreign key
21	extend runit :
22	remove foreign key;
23	include faglflex01_data not null;
24	include faglflex02_inc not null;
25	include faglflex03_data not null;
26	include ci_faglflex04 not null;
27	include ci_faglflex05 not null;
28	include faglflex06_inc not null;
29	value_data : include faglflex_si_value_data;
30	fix : include faglflex_si_fix_inc not null;
31	include ci_faglflexa not null;
32	_dataaging : data_temperature;
33	
34	}

Abbildung 2.12 Anzeige der Kompatibilitäts-View-Definition innerhalb der Tabellendefinition

Der CDS View FGL_FAGLFLEXA nutzt eine tief verschachtelte CDS-View-Aufrufhierarchie, das heißt, er ruft einen CDS View auf, der einen weiteren CDS View konsumiert usw. Wenn man dieser Aufrufhierarchie folgt, gelangt man letztendlich zur Datenermittlung aus den neuen Tabellen ACDOCA, FINSC_LEDGER_REP und FINSC_LD_CMP. Hierüber ist nachvollziehbar, wie die Daten aus der ehemaligen transparenten Tabelle in SAP S/4HANA über das neue Datenmodell ermittelt werden.

Da die transparente Tabelle aus technischer Sicht durch einen CDS View ersetzt wurde, hat diese Änderung unter gewissen Umständen Auswirkungen auf die Eigenentwicklungen. Beispielsweise sind schreibende SQL-Operationen auf einen CDS View wie INSERT, UPDATE, MODIFY oder DELETE grundsätzlich nicht möglich.

Kompatibilitäts-Views V1 und Kompatibilitäts-Views V2 reagieren innerhalb der ABAP-Entwicklung teilweise unterschiedlich. Wir veranschaulichen das Systemverhalten am Beispiel des Kompatibilitäts-Views V1 für den Zugriff auf Daten aus Tabelle FAGLFLEXT und des Kompatibilitäts-Views V2 für den Zugriff auf Daten aus FAGLFLEXA.

Bei der Aktivierung des Codings aus Listing 2.1 kommt es in SAP S/4HANA zu einem Syntaxfehler (»Der Datenbank-View "FAGLFLEXT" ist schreibgeschützt und kann deshalb nicht verändert werden«). Deshalb wird es bei der Ausführung des Codings zu einem Laufzeitfehler kommen.

```

UPDATE faglflexxt FROM ls_faglflexxt.
WRITE: 'sy-subrc:', sy-subrc.

```

Listing 2.1 SQL-Update-Anweisung auf einen Kompatibilitäts-View V1

Anders sieht es mit dem Beispiel in Listing 2.2 aus.

```

UPDATE faglflexa FROM ls_faglflexa.
WRITE: 'sy-subrc:', sy-subrc.

```

Listing 2.2 SQL-Update-Anweisung auf einen Kompatibilitäts-View V2

Die Aktivierung und technische Ausführung des Codings ohne Laufzeitfehler ist grundsätzlich noch möglich. Das Ergebnis in der Variablen sy-subrc aus Listing 2.2 wird nach der SQL-Update-Anweisung immer den Wert »4« haben, da ein Update über den CDS View technisch nicht möglich ist.

Die Einschränkungen bei der Verwendung von Kompatibilitäts-Views beziehen sich nicht nur auf ändernde SQL-Operationen, sondern können auch dazu führen, dass Anpassungen bei der Datenselektion Ihrer Anwendungen erforderlich werden. In Listing 2.3 werden alle Daten aus dem Kom-

patibilitäts-View V1 FAGLFLEXT ermittelt, in einer lokalen Tabelle gespeichert und nach dem Primärschlüssel sortiert. Unter SAP S/4HANA verursacht die Ausführung dieser SQL-Anweisung einen Laufzeitfehler (Dump).

```
SELECT * FROM faglflect
      INTO TABLE @DATA(it_faglflect)
      ORDER BY PRIMARY KEY.
```

Listing 2.3 Ungültige Verwendung von ORDER BY am Beispiel des Kompatibilitäts-Views V1 FAGLFLEXT

Zur Korrektur des Laufzeitfehlers muss die explizite Feldliste in der ORDER BY-Anweisung erweitert werden. Listing 2.4 zeigt eine korrigierte Variante des in Listing 2.3 gezeigten Aufrufes.

```
SELECT * FROM faglflect
      INTO TABLE @DATA(it_faglflect)
      ORDER BY rclnt, ryear, objnroo.
```

Listing 2.4 Korrigierte Verwendung von ORDER BY am Beispiel des Kompatibilitäts-Views V1 FAGLFLEXT

Die ORDER BY-Anweisung wird bei Kompatibilitäts-Views V2 vollständig unterstützt.

2.3.3 Die Top 10 der Simplification Items

Häufig auftretende Befunde

Beim Custom Code Check treten immer wieder bestimmte Befunde mit der gleichen Prüfmeldung auf, unabhängig von der Anzahl oder Komplexität der Eigenentwicklungen oder von der Branchenzuordnung des Zielsystems. Die in diesem Abschnitt zusammengestellte Top 10 der Simplification Items soll Ihnen nicht nur einen Überblick über häufig auftretende Befunde geben, sondern zu jedem Befund auch Details und Hintergründe klären. Ferner geben wir Ihnen jeweils eine allgemeine Empfehlung zu notwendigen Erweiterungen.

Nr. 1: Feldlängenerweiterung für Materialnummern (SAP-Hinweis 215424)

Mit SAP S/4HANA 1511 wurde die Domäne für die Materialnummer von 18 auf 40 Stellen erweitert. Diese technische Erweiterung wird bei einer Systemkonvertierung nach SAP S/4HANA über den SUM durchgeführt. Jedes SAP-S/4HANA-System arbeitet damit automatisch mit einer 40-stelligen Materialnummer auf Basis der Domäne MATNR. Abbildung 2.13 zeigt das entsprechende Datenelement MATNR mit der zugeordneten Domäne MATNR im ABAP Dictionary.

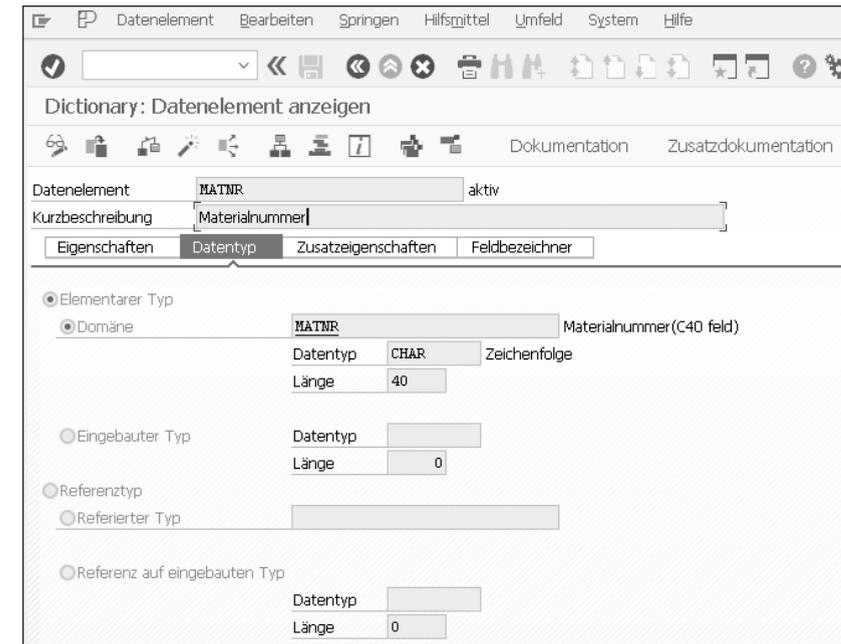


Abbildung 2.13 Datenelementdefinition von MATNR in SAP S/4HANA 1909

Ausnahme SAP S/4HANA Finance 1605

Auf der ersten Etappe zu SAP S/4HANA 1503, SAP Simple Finance, betrug die maximale Feldlänge der Materialnummer noch 18 Zeichen. Mit SAP S/4HANA 1511 wurde die Materialnummer von 18 auf 40 Stellen erweitert. Da SAP S/4HANA Finance 1605 auf SAP ERP 6.0 EHP 8 basiert, wird hier noch die 18-stellige Materialnummer verwendet. Ab SAP S/4HANA 1610 bleibt die Domänenenerweiterung auf eine 40-stellige Materialnummer auch für die darauffolgende Releases bestehen.

Über das Customizing kann die Verwendung der 40-stelligen Materialnummer aktiviert werden. Ohne diese Aktivierung kann in SAP S/4HANA ebenso wie in SAP ERP nur die 18-stellige Materialnummer verwendet werden. Die deaktivierte Feldlängenerweiterung ist die Standardeinstellung im Customizing nach einer Systemkonvertierung oder einer Neuinstallation.

Um die 40-stellige Materialnummer zu verwenden, können Sie diese in Transaktion SPRO über folgenden Pfad aktivieren:

SAP Referenz-IMG • Anwendungsübergreifende Komponenten • Allgemeine Anwendungsfunktionen • Feldlängenerweiterung • Erweiterte Felder aktivieren



Aktivierung im Customizing

Alternativ können Sie die mandantenabhängige Customizingtabelle auch über die Transaktion FLETS aufrufen.

Um die Aktivierung durchzuführen, wechseln Sie in den Bearbeitungsmodus (☰) und markieren die Checkboxen **Materialnummernerweiterung** und **Bestätige, dass ich SAP-Hinweis 2232396 gelesen habe** (siehe Abbildung 2.14). Speichern Sie diese Änderung. Anschließend können Sie das Materialnummernformat über die Transaktion OMSL kundenspezifisch anpassen, um eine Feldlänge von mehr als 18 Zeichen zu unterstützen.

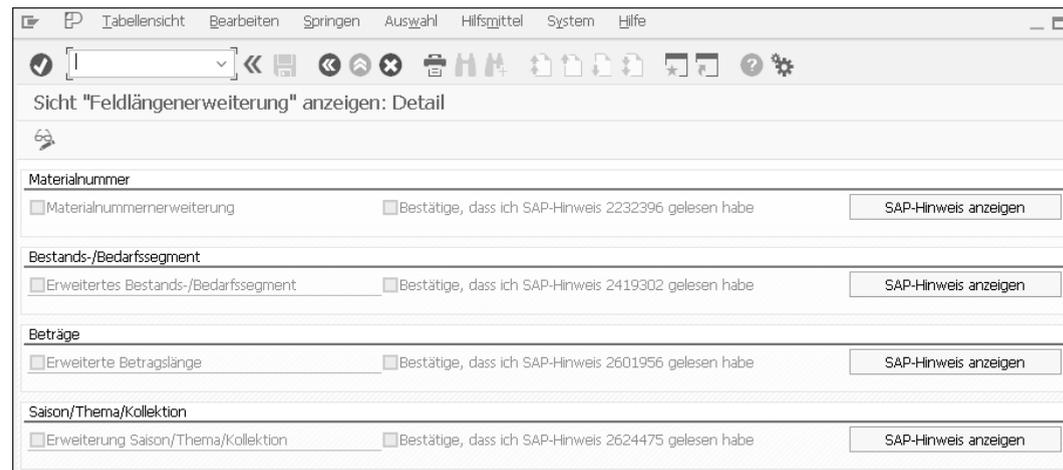


Abbildung 2.14 Aktivierung der Feldlängenerweiterung im Customizing



Keine Deaktivierung der Feldlängenerweiterung

Sobald die Feldlängenerweiterung in einem System aktiviert wurde, ist eine nachträgliche Deaktivierung in der Regel nicht mehr möglich. Das System geht davon aus, dass nach der Aktivierung bereits Datenbestände verbucht wurden, die die Feldlängenerweiterung verwenden. Um Dateninkonsistenzen zu vermeiden bzw. den korrekten Zugriff auf die verbuchten Daten weiterhin zu gewährleisten, wird deshalb eine nachträgliche Deaktivierung nicht empfohlen, auch wenn diese über die Transaktion FLETS ausgeführt werden kann.

Empfehlung zur Aktivierung

Wir empfehlen eine Aktivierung der erweiterten Materialnummer nur dann, wenn die zuvor verwendeten 18 Stellen in absehbarer Zeit nicht mehr ausreichen. Eine rein vorsorgliche Aktivierung der erweiterten Materialnummer ist zwar auch möglich, die daraus resultierenden Aufwände sind jedoch nicht unerheblich und sollten bei dieser Entscheidung bedacht werden. Innerhalb von Eigenentwicklungen können Konstellationen auftreten, die mit einer

40-stelligen Materialnummer nicht mehr funktionsfähig sind, weshalb diese Entwicklungen umgestellt werden müssen. Auch kann es bei einer Multisystemlandschaft zu Inkompatibilitäten innerhalb der Kommunikation kommen, wenn andere Schnittstellensysteme die 40-stellige Materialnummer nicht unterstützen. Der SAP-Hinweis 2232396 gibt einen aktuellen Überblick zu Integration und Einschränkungen mit anderen SAP-Business-Suite-Systemen für die aktivierte Längenerweiterung der Materialnummer.



Auswirkungen auf andere Domänen

Die Materialnummer-Feldlängenerweiterung hat auch Auswirkungen auf andere Objekte. Beispielsweise wurde die Domäne für Klassifikationsmerkmale (Datenelement ATWRT) in SAP S/4HANA von 30 auf 70 Stellen erweitert, unter anderem auch, um dort bei der Bewertung die 40-stellige Materialnummer hinterlegen zu können. Die Verwendung des 70-stelligen Klassifikationsmerkmals ist jedoch nur nach Aktivierung der Materialnummer-Feldlängenerweiterung möglich.

Verglichen mit einem SAP-ERP-System hat sich das Format, wie die Materialnummer in der Datenbank gespeichert wird, nicht geändert. Die Materialnummer wird in der Datenbank weiterhin wie gewohnt linksbündig, gemäß dem im Customizing definierten Materialnummernformat, gespeichert. Deshalb ist in der Regel auch keine Datenumsetzung notwendig, wenn von SAP ERP nach SAP S/4HANA konvertiert wird. Tabelle 2.3 zeigt die Speicherung der Materialnummer auf Ebene der Datenbank bei einer 18-stelligen Materialnummer in SAP ERP und einer 40-stelligen in SAP S/4HANA. In beiden Fällen wird die Materialnummer linksbündig auf Basis des definierten Materialnummernformats abgelegt.

Speicherung in der Datenbank

Datentyp	MATNR (Länge 18) in SAP ERP	MATNR (Länge 40) in SAP S/4HANA
numerisch	000000000000000123	000000000000000123
alphanumerisch	MAT123	MAT123

Tabelle 2.3 Speicherung der Materialnummer auf Ebene der Datenbank in SAP ERP und SAP S/4HANA

In SAP S/4HANA wurde bei vielen Funktionsbausteinen, Klassenmethoden, Business Add-ins (BADIs) usw. die Signatur bzw. Implementierung angepasst, um die erweiterte Materialnummer zu unterstützen. Bei lokalen/nicht remotefähigen Funktionsbausteinen wurde in der Regel lediglich sicherge-

Erweiterung von Entwicklungsobjekten

stellt, dass statt der 18-stelligen Materialnummer nun eine 40-stellige Materialnummer unterstützt wird. Auch bei nicht freigegebenen, remotefähigen Funktionsbausteinen, die in der Regel nur systemintern verwendet werden, wurde diese Erweiterung durchgeführt. Die Signatur dieser Funktionen ist meist gleich geblieben.

Neue Parameter

Bei freigegebenen, remotefähigen Schnittstellen wurde ein anderes Vorgehen gewählt. Die Signatur der Funktionen wurde um ein oder mehrere Felder für die 40-stellige Materialnummer erweitert. Andernfalls könnte es zu Kommunikationsproblemen bei bestehenden Schnittstellen nach dem Umstieg auf SAP S/4HANA kommen. Die technische Definition von bestehenden Parametern bleibt deshalb unverändert. Die 40-stellige Materialnummer wurde bei solchen Funktionen lediglich in Form von zusätzlichen Parametern ergänzt.

In Abbildung 2.15 und Abbildung 2.16 wird SAPs Vorgehen bei dieser Erweiterung am Beispiel des Funktionsbausteins BAPI_MATERIAL_GETLIST veranschaulicht. Dieser freigegebene remotefähige Funktionsbaustein liefert auf Basis der Tabelle MATNRSELECTION alle Materialstammdaten in der Tabelle MATNRLIST zurück.

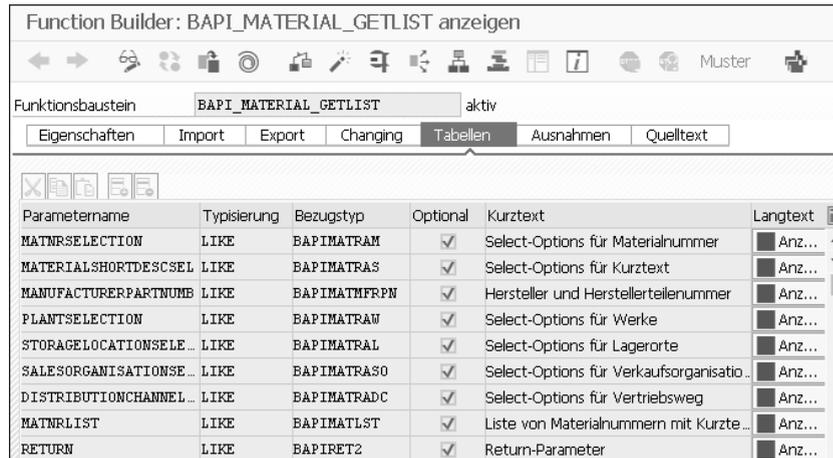


Abbildung 2.15 Signatur des Funktionsbausteins BAPI_MATERIAL_GETLIST in SAP S/4HANA

Die in Abbildung 2.16 dargestellten Parameter in der Tabelle MATNRSELECTION für die Felder MATNR_LOW und MATNR_HIGH haben in SAP S/4HANA ebenfalls eine technische Länge von 18 Stellen für die Materialnummer. Wird dieser Funktionsbaustein von einem externen System aufgerufen und wird dabei weiterhin die 18-stellige Materialnummer in diesen Feldern übergeben, ist eine korrekte Verarbeitung auch in SAP S/4HANA möglich.

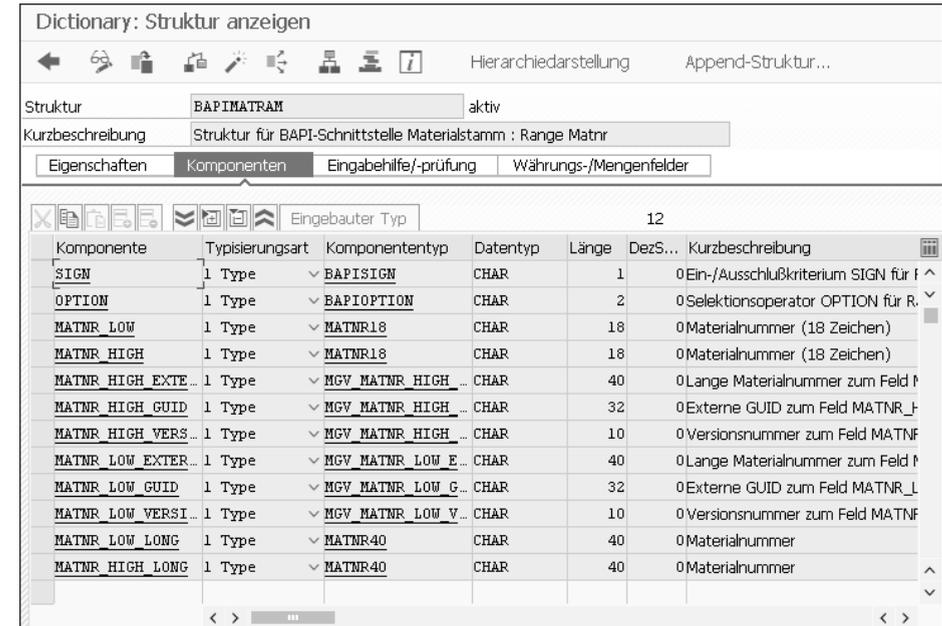


Abbildung 2.16 Struktur BAPIMATRAM zur Tabelle MATNRSELECTION

Sofern das Zielsystem eine 40-stellige Materialnummer unterstützt, können und sollten die neuen *_LONG-Felder verwendet werden. Diese Empfehlung gilt auch, wenn der Funktionsbaustein nur lokal aufgerufen wird. Auch wenn Sie die erweiterte Materialnummer über das Customizing zum aktuellen Zeitpunkt nicht aktivieren, wird bei ausschließlich lokalen Aufrufen empfohlen, das Coding auf die neuen *_LONG-Felder umzustellen, um eine spätere Kompatibilität zu gewährleisten.

SAP-Hinweis 2438131 beschreibt die Hintergründe der Erweiterungen, die bei lokalen oder Remote-Aufrufen von Funktionsbausteinen in SAP S/4HANA beachtet werden sollten. Wenn Sie innerhalb einer Eigenentwicklung auf einen so geänderten Funktionsbaustein zugreifen, liefert der Custom Code Check (CCC) in der Regel einen entsprechenden Befund. Dieser Befund wird direkt in der Zeile des Funktionsaufrufes (CALL FUNCTION) gemeldet. Der Prüftitel gibt Auskunft darüber, um welche Art der Änderung es sich handelt (z. B. Materialfeldlängenerweiterung). Jedoch liefert der Befund keine Detailinformationen dazu, welche Übergabe- oder Rückgabestruktur von der Änderung betroffen ist. Hilfreich ist hier die PDF-Datei in den Anlagen des genannten SAP-Hinweises. Sie bietet einen sehr guten Überblick, bei welchem Funktionsbaustein in welcher technischen Struktur eine Erweiterung für die Materialnummer durchgeführt wurde. Dies ist sehr hilfreich, da die von SAP erweiterten Funktionsbausteine sehr umfangreich sein können.

Verwendung auch bei lokalen Aufrufen

Bearbeitung der CCC-Befunde

Im Kontext der Materialnummer-Feldlängenerweiterung sind neben etwaigen Erweiterungen von Dynpros, innerhalb von Schnittstellen, Formularen usw. noch viele weitere Punkte zu beachten, die jedoch den Rahmen dieses Abschnitts sprengen würden. SAP-Hinweis 2267140 gibt einen guten Überblick zu weiteren relevanten Abhängigkeiten. Das PDF-Dokument in den Anlagen dieses Hinweises fasst die wichtigsten Punkte mit mehreren Beispielen sehr gut zusammen.

Nr. 2: Feldlängenerweiterung für Beträge (SAP-Hinweis 2610650)

Um den Anforderungen von Banken und Finanzinstituten gerecht zu werden, wurde die Länge des Betragsfeldes in den Anwendungsbereichen für das Hauptbuch und im Controlling erweitert. Ab SAP S/4HANA 1809 wurden Währungsbetragsfelder mit einer Feldlänge zwischen 9 und 22 Stellen einschließlich zwei Dezimalstellen auf 23 Stellen erweitert. Neben diesen Währungsbetragsfeldern sind ausgewählte Datenelemente (Typ DEC, CHAR und NUMC) mit verschiedenen Längen und Dezimalstellen, die Beträge enthalten, ebenfalls von dieser Erweiterung betroffen.

Im Folgenden veranschaulichen wir diese Erweiterung an Hand des Datenelements DMBTR. Abbildung 2.17 zeigt die Definition des Datenelements DMBTR in SAP ERP. Der Datentyp ist hier der für Währungen (CURR) mit einer Länge von 13 Stellen und zwei Dezimalstellen. Als Domäne verwendet dieses Datenelement WERT7.

Abbildung 2.17 Definition des Datenelements DMBTR in SAP ERP

In Abbildung 2.18 ist die gleiche Datenelementdefinition in SAP S/4HANA 1909 zu sehen. Die Domäne hat den Namen AFLE13D2016N_TO_23D2030N und die technische Länge von 23 Stellen.

Abbildung 2.18 Definition des Datenelements DMBTR in SAP S/4HANA 1909

Wenn Sie den elementaren Datentyp hinter der Domäne in Abbildung 2.17 und Abbildung 2.18 vergleichen, erkennen Sie, dass das Datenelement in SAP S/4HANA um zehn Stellen erweitert wurde.

Namenskonventionen neuer Domänen

Der Domänenname AFLE13D2016N_TO_23D2030N in SAP S/4HANA sieht auf den ersten Blick kryptisch aus, folgt aber einer strikten Namenskonvention. Die Namenskonvention verschlüsselt die Länge, die Nachkommastellen, die Ausgabelänge und die Vorzeicheninformation der Ersetzungs- und Zieldomäne. Bei der Domäne AFLE13D2016N_TO_23D2030N hat die Quelldomäne beispielsweise eine Länge von 13 Stellen und zwei Nachkommastellen, eine Ausgabelänge von 16 und kein Vorzeichen. Daraus ergibt sich der Namensbestandteil 13D2016N. Die Zieldomäne hat eine Länge von 23 Stellen und zwei Nachkommastellen, eine Ausgabelänge von 30 und kein Vorzeichen, woraus sich der Namensbestandteil 23D2030N ergibt. Weitere Regeln für diese Namenskonvention sind in SAP-Hinweis 2628040 beschrieben.

Analog zur im vorangegangenen Abschnitt beschriebenen Materialnummer-Feldlängenerweiterung wurde auch hier die Domäne der bereits exist-

Auswirkungen auf ABAP-Coding

tierenden Datenelemente angepasst. Bei diesem Ansatz ist ABAP-Coding, das die betroffenen Typen verwendet, in den meisten Fällen nach wie vor syntaktisch korrekt. Unter Umständen kann es jedoch zu ungewollten Seiteneffekten kommen, beispielsweise wenn in der Ausgabe eines Dynpros von maximal 13 Stellen ausgegangen wird und deshalb der Wert abgeschnitten wird.

In seltenen Fällen kann es aber auch zu Laufzeitfehlern kommen. Eine der häufigsten Ursachen ist eine Inkompatibilität zwischen Aktualparameter und Formalparameter, das heißt, wenn der Parametertyp, der einer Funktion übergeben wird, nicht mit der Funktionssignaturdefinition übereinstimmt. Auch können in Einzelfällen Überlauferfehler auftreten, wenn ein erweiterter Betrag in ein kürzeres Betragesfeld verschoben wird.

Aktivierung über das Customizing

Analog zur Materialnummer-Feldlängenerweiterung kann auch die Erweiterung des Betragesfeldes über die Transaktion FLETS im Customizing aktiviert werden. Auch hier ist keine nachträgliche Deaktivierung mehr möglich, nachdem die Aktivierung einmal erfolgt ist.

Unabhängig davon, ob diese Feldlängenerweiterung aktiviert wird oder nicht, sollten Sie Ihre Eigenentwicklungen so weit wie möglich dahingehend anpassen, dass die maximale Betragesfeldlänge von 23 Stellen in den Währungsfeldern über die neuen Domänen unterstützt wird. Diese Anpassung dient der Sicherstellung einer späteren Kompatibilität der Eigenentwicklungen. Die technische Erweiterung der Domäne ist genau wie bei den Materialnummer-Feldlängen automatisch gegeben, sobald die Systemkonvertierung nach SAP S/4HANA durchgeführt wurde. Sie ist unabhängig von der Aktivierung über das Customizing. Ihre Eigenentwicklungen sollten Sie deshalb in jedem Fall überprüfen und gegebenenfalls anpassen.

Ersetzungsdatenelemente

Für alle erweiterten Datenelemente wurden von SAP Kopien der Originaldefinitionen unter denselben Namen, jedoch mit dem Suffix `_CS` als sogenannte *Ersetzungsdatenelemente* angelegt (siehe z. B. das Ersetzungsdatenelement `DMBTR_CS` in Abbildung 2.19). Ist die Unterstützung des neuen 23-stelligen-Datenelements innerhalb einer Eigenentwicklung nicht möglich oder explizit nicht gewünscht, sollte die Umstellung auf diese Ersetzungsdatenelemente hin geprüft werden.

Unterstützung der Betragesfeld-erweiterung

Ab SAP S/4HANA 1809 werden die Ersetzungsdatenelemente in Standardprozessen bereits überall dort verwendet, wo eine Unterstützung der erweiterten Betrageslänge nicht geplant ist (z. B. in der Logistik). Nur bestimmte Funktionen oder Applikationen mit Fokus auf die Anwendungsbereiche FI/CO unterstützen die erweiterte Betrageslänge nach erfolgter Aktivierung. Dazu gehören z. B. die Anwendungen »Sachkonten ausgleichen«, »Hauptbuchbelege prüfen«, »Vorerfasste Buchungsbelege anzeigen« oder »Profit-

center verwalten«. Eine Auflistung der Funktionen, die eine erweiterte Betragesfeldlänge unterstützen, können Sie dem SAP-Hinweis 2628654 entnehmen.

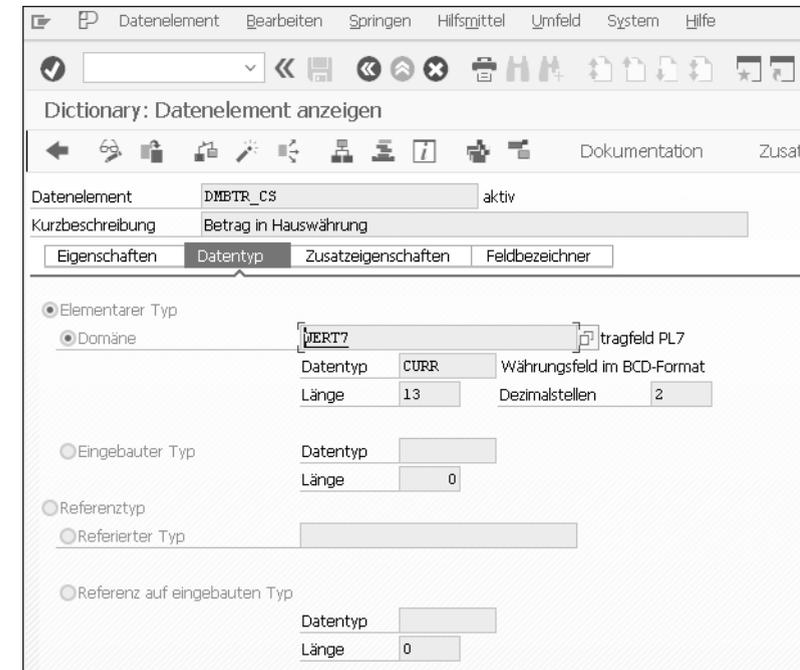


Abbildung 2.19 Ersetzungsdatenelement `DMBTR_CS` in SAP S/4HANA 1909

Nr. 3: Datenmodelländerungen in der Preisfindung und Konditionstechnik (SAP-Hinweis 2220005)

Das Ergebnis zur Preisfindung für Aufträge oder Bestellungen wird in SAP ERP in der Tabelle `KONV` gespeichert. In SAP S/4HANA wurde die Tabelle `KONV` durch die neue Datenbanktabelle `PRCD_ELEMENTS` ersetzt. In der Tabelle `KONV` werden keine Daten mehr gespeichert. Ein automatischer Zugriff auf die Tabelle `PRCD_ELEMENTS` über einen Kompatibilitäts-View wird bei einer `SELECT`-Anweisung auf die Tabelle `KONV` *nicht* unterstützt. Jeder Zugriff auf die Tabelle `KONV` in Eigenentwicklungen muss deshalb überprüft und gegebenenfalls angepasst werden, unabhängig davon, ob es sich um einen lesenden oder ändernden Zugriff handelt.

Grundsätzlich wird empfohlen, Datenzugriffe so weit wie möglich über APIs durchzuführen. SAP bietet dazu die Klasse `CL_PRC_RESULT_FACTORY` für den lesenden und schreibenden Zugriff an. Darüber hinaus kann für komplexere Selektionen auch der CDS View `V_KONV` verwendet werden. Dieser liest die Daten aus der Tabelle `PRCD_ELEMENTS` im Format der Tabelle `KONV`.

Factory-Klasse für
`PRCD_ELEMENTS`



Performanceeinschränkungen bei Verwendung von V_KONV

Aus Performancegründen empfiehlt SAP, den CDS View V_KONV nicht für einfache Selektionen zu verwenden. Stattdessen sollte das Coding auf einen API-Zugriff umgestellt werden. Die von SAP ausgelieferten APIs haben häufig einen automatischen Pufferungsmechanismus. Da die Tabelle PRCD_ELEMENTS als transparente Tabelle ohne Pufferung erstellt wurde, sind direkte Datenbankzugriffe im Vergleich zu API-Zugriffen unter Umständen langsamer.

In Abbildung 2.20 sehen Sie die Definition der Tabelle KONV aus einem SAP-ERP-System und in Abbildung 2.21 die Definition der Tabelle PRCD_ELEMENTS in SAP S/4HANA 1909.

Die wichtigsten Änderungen der Tabelle PRCD_ELEMENTS im Vergleich zur Tabelle KONV lassen sich wie folgt zusammenfassen:

- Der Schlüssel ist in der neuen Tabelle unverändert.
- Bestimmte Spalten wurden entfernt (z. B. WEGXX und STUFE).
- Bei anderen Spalten wurde eine Felderweiterung durchgeführt (z. B. KRECH, KAWRT, KBETR, KWERT, KOPOS).
- Bestimmte Spalten wurden zusammengeführt (KONV-KOLNR und KONV-KOLNR3 in PRCD_ELEMENTS-KOLNR).
- Eine neue Spalte wurde eingeführt (WAERK).

Feld	Key	Ini...	Datenelement	Datentyp	Länge	DezS...	Kurzbeschreibung
MANDT	✓	✓	MANDT	CLMT	3	0	0Mandant
KNUMV	✓	✓	KNUMV	CHAR	10	0	0Nummer der Belegkondition
KPOSN	✓	✓	KPOSN	NUMC	6	0	0Konditionspositionsnummer
STUNR	✓	✓	STUNR	NUMC	3	0	0Stufennummer
ZAEHK	✓	✓	DZAEHK	NUMC	2	0	0Zähler Konditionen
KAPPL			KAPPL	CHAR	2	0	0Applikation
KSCHL			KSCHA	CHAR	4	0	0Konditionsart
KDATU			KDATU	DATS	8	0	0Konditions-Preisstellungsdatum
KRECH			KRECH	CHAR	1	0	0Konditions-Rechenregel
KAWRT			KAWRT	CURR	15	2	2Konditionsbasis
KBETR			KBETR	CURR	11	2	2Konditionsbetrag oder -prozentsatz
WAERS			WAERS	CUKY	5	0	0Währungsschlüssel
KKURS			KKURS	DEC	9	5	5Konditionskurs für Umrechnung in Hausv

Abbildung 2.20 Datenbanktabelle KONV in SAP ERP

Feld	Key	Ini...	Datenelement	Datentyp	Länge	DezS...	Koordinat...	Kurzbeschreibung
CLIENT	✓	✓	MANDT	CLMT	3	0	0	0Mandant
.INCLUDE	✓	✓	PRCS_ELEMENTS_K	STRU	0	0	0	
KNUMV	✓	✓	KNUMV	CHAR	10	0	0	0Nummer der Belegkonditio
KPOSN	✓	✓	KPOSN	NUMC	6	0	0	0Konditionspositionsnumme
STUNR	✓	✓	STUNR	NUMC	3	0	0	0Stufennummer
ZAEHK	✓	✓	VFPRC_COND_COUNT	NUMC	3	0	0	0Konditionszähler
.INCLUDE			PRCS_ELEMENTS_D	STRU	0	0	0	
KAPPL			KAPPL	CHAR	2	0	0	0Applikation
KSCHL			KSCHA	CHAR	4	0	0	0Konditionsart
KDATU			VFPRC_TIMESTAMP	CHAR	14	0	0	0Zeitstempel für Preisfindur
KRECH			KRECH_LONG	CHAR	3	0	0	0Konditionsrechenregel
KAWRT			VFPRC_ELEMENT_B	DEC	24	9	0	0Konditionsbasiswert

Abbildung 2.21 Datenbanktabelle PRCD_ELEMENTS in SAP S/4HANA

Zusätzlich wurde das Datenmodell in den Konditionstabellen dahingehend angepasst, dass der *variable Schlüssel* (VAKEY) und das *variable Datum* (VADAT) aus allen Konditionskopftabellen entfernt wurden. Diese Änderung betrifft auch z. B. die Tabellen KONH (Preisfindung), NACH (Nachrichtensteuerung), KONDB3 (Kampagnenfindung), KONDN (Naturalrabatt), KONHM (Portfoliofindung), J_3GPRLHD (ETM Preislisten Konditionskopf) und WIND (Belegindex). Für die interne Verarbeitung wurden als Ersatz die neuen Datenelemente VAKEY_LONG und VADAT_KO_LONG eingeführt, auf die in SAP S/4HANA über die neue Serviceklasse CL_COND_VAKEY_SRV zugegriffen werden kann.

Eliminierung von VAKEY und VADAT

Nr. 4: Geschäftspartneransatz (SAP-Hinweis 2265093)

Ab SAP S/4HANA 1511 ist der Geschäftspartner das neue führende Objekt zur Pflege von Kunden- und Lieferantendaten. In SAP ERP wurden Kunden und Lieferanten noch als getrennte, autarke Objekte behandelt. In SAP S/4HANA ist die Pflege der Kunden- und Lieferanten-Stammdaten zentral über die Geschäftspartner-Transaktion BP zugänglich. Die alten Transaktionen zur Anzeige bzw. Pflege des Kunden- bzw. Lieferantenstamms, wie beispielsweise VDO2, XDO1 und FDO3 können in SAP S/4HANA nicht mehr verwendet werden. Beim Aufruf dieser Transaktionen werden Sie automatisch an die Transaktion BP weitergeleitet.



Aktivierung des Geschäftspartneransatzes

Der Geschäftspartner ist in SAP Simple Finance 1503 und SAP S/4HANA Finance 1605 noch optional, wird aber ab SAP S/4HANA 1511 obligatorisch. Sie können den Geschäftspartner bereits in SAP ERP aktivieren und verwenden. Die Einführung des Geschäftspartners kann als ein mögliches Vorprojekt zur SAP-S/4HANA-Migration eingeplant werden.

In SAP ERP haben Sie die Wahl, ob Sie weiterhin die alten Debitoren-/Kreditoren-Transaktionen nutzen möchten, damit Ihre Stammdaten mit dem Geschäftspartner synchronisiert werden. Eine umgekehrte Aktivierung der Synchronisation mit den Debitoren- und Kreditoren-Stammdaten über die Transaktion BP ist in SAP ERP ebenfalls möglich.

Customer-Vendor-Integration

Aus Kompatibilitätsgründen werden die alten Datenbanktabellen für den Kundenstamm (KNA1, KNVV usw.) oder Lieferantenstamm (LFA1, LFB1 usw.) über die sogenannte *Customer-Vendor-Integration* (CVI) mit den Geschäftspartnerstammdaten synchronisiert. Kundeneigene Felder an diesen Stammdatentabellen bleiben nach der Systemkonvertierung unverändert. Da häufig auch Standardtransaktionen erweitert wurden, um die Anzeige und Pflege von kundeneigenen Feldern zu unterstützen, ist eine zusätzliche analoge Erweiterung der Transaktion BP in SAP S/4HANA erforderlich. Die Erweiterung dieser Transaktion kann in der Regel nicht automatisch erfolgen, da häufig konzeptionelle Vorarbeiten erforderlich sind, wie die Definition der relevanten Geschäftspartnerrollen, einer Registerkarte, eines Abschnitts, einer Berechtigung usw.

Nr. 5: Datenmodelländerungen im Vertrieb (SAP-Hinweis 2198647)

In der Vertriebskomponente SD wurden folgende Änderungen durchgeführt:

■ Beseitigung der Statustabellen VBUK und VBUP

Die Tabellen VBUK (Statusinformation für SD-Belegkopf) und VBUP (Statusinformationen für SD-Belegposition) existieren in SAP S/4HANA zwar noch, werden aber nicht mehr fortgeschrieben. Im Zuge der technischen Systemkonvertierung werden die Statusinformationen aus diesen beiden Tabellen in das neue SAP-S/4HANA-Datenmodell überführt. Im neuen Datenmodell werden die Statusinformationen direkt in den Tabellen zum SD-Belegkopf bzw. der SD-Belegposition gespeichert. Technisch werden die Statusinformationen zu SD-Aufträgen nun direkt in den Tabellen VBAK und VBAP fortgeschrieben, für Lieferungen in den Tabellen LIKP und LIPS und für Fakturen in der Tabelle VBRK.

■ Feldlängenerweiterung für den Vertriebsbelegtyp (VBTP)

Der Vertriebsbelegtyp wurde von einem einstelligen Zeichenfeld mit dem Datenelement VBTP zu einem vierstelligen Zeichenfeld mit dem Datenelement VBTPPL erweitert.

Anstatt den Vertriebsbelegtyp über statische Literale abzufragen (z. B. C für »Auftrag«), wird die Verwendung der Klasse IF_SD_DOC_CATEGORY empfohlen. Sie besitzt viele statische Konstanten, die direkt verwendet werden können (für Aufträge z. B. IF_SD_DOC_CATEGORY=>ORDER).

■ Änderung der technischen Struktur der Belegflusstabelle VBFA

In früheren SAP-S/4HANA-Versionen wurden in der Belegfluss-Tabelle VBFA nur direkte Belegbeziehungen gespeichert. Ab SAP S/4HANA 1709 werden darin auch indirekte Belegbeziehungen gespeichert. Hierzu wurde die Tabelle VBFA erneut um das Feld STUFE erweitert. Der zusammengesetzte Primärschlüssel mit Vorgängerbeleg, Folgebeleg und zugehörigem Vertriebsbelegtyp wurde durch den neuen Primärschlüssel RUUID vom Typ RAW(16) ersetzt.

Der Zugriff auf die Tabelle über Vorgängerbeleg, Folgebeleg und Vertriebsbelegtyp ist nach wie vor möglich. Da der Vertriebsbelegtyp erweitert wurde, sind die Felder VBTYPEXT_V und VBTYPEXT_N nicht mehr notwendig und wurden deshalb entfernt. Für den Zugriff auf die Tabelle VBFA wird die Verwendung der Klasse CL_SD_DOCUMENT_FLOW_RT empfohlen, sofern dies in der betreffenden Zugriffskonstellation möglich ist.

■ Eliminierung der Indextabellen für SD-Belege nach Material und Kunde

Um einen performanten Zugriff auf SD-relevante Informationen zu gewährleisten, wurden in speziellen Tabellen redundante Daten parallel zu den Originaltabellen fortgeschrieben. Da mit SAP HANA ein performanter Zugriff direkt auf die Originaltabellen möglich ist, wurden in SAP S/4HANA die folgenden Tabellen entfernt:

- VAKPA (Vertriebsindex: Aufträge zu Partnerrollen)
- VAPMA (Vertriebsindex: Auftragspositionen zu Material)
- VLKPA (Vertriebsindex: Lieferungen zu Partnerrollen)
- VLPMA (Vertriebsindex: Lieferungspositionen zu Material)
- VRKPA (Vertriebsindex: Fakturen zu Partnerrollen)
- VRPMA (Vertriebsindex: Fakturapositionen zu Material)

In SAP S/4HANA werden CDS Views für diese Tabellen ausgeliefert, deren Namen sich aus dem V_ und dem alten Tabellenamen zusammensetzen, z. B. V_VAKPA. Diese ermitteln die Daten in der alten Struktur, jedoch direkt aus den neuen Originaltabellen. Aus Kompatibilitätsgründen wurden diese CDS Views als Kompatibilitäts-Views V2 eingeführt. Damit funktio-

nieren automatisch alle SELECT-Anweisungen auf die alten Tabellen wie zuvor. Ändernde Operationen sind technisch nicht möglich und müssen, wie bei Kompatibilitäts-Views üblich, umgestellt werden.

Nr. 6: Datenmodelländerungen in der Anlagenbuchhaltung (SAP-Hinweis 2270387)

Für die neue Anlagenbuchhaltung ist die Aktivierung der neuen Hauptbuchhaltung obligatorisch. Bei der neuen Anlagebuchhaltung werden die Ist-Werte statt in den Tabellen ANEP (Anlagen-Einzelposten), ANEA (Anlagen-Einzelposten anteilige Werte), ANLP (Anlagen-Periodenwerte) und ANLC (Anlagen-Wertfelder) nun direkt in der Tabelle ACDOCA (Universal Journal) gespeichert. Die Werte aus der Tabelle ANEK (Belegkopf Anlagenbuchung) werden in den Tabellen BKPF und ACDOCA gespeichert. Die Tabelle BSEG (Belegsegment Buchhaltung) wird weiterhin wie gewohnt fortgeschrieben.

Durch die Integration von Kompatibilitäts-Views können lesende Zugriffe auf die Datenbanktabellen in der Regel wie gewohnt genutzt werden, schreibende Zugriffe müssen jedoch umgestellt werden. Statistische Werte (z. B. für Steuerzwecke) aus den Tabellen ANEP, ANEA, ANLP und ANLC werden in der neuen Tabelle FAAT_DOC_IT (Statistische Einzelposten in der Anlagenbuchhaltung) fortgeschrieben. Die Planwerte aus den Tabellen ANLP und ANLC werden in der Tabelle FAAT_PLAN_VALUES (Geplante Abschreibungen und Aufwertungen) gespeichert.

Nr. 7: Außenhandel in SD/MM (SAP-Hinweis 2223144)

In SAP S/4HANA sind die Außenhandelsfunktionen (SD-FT oder MM-FT) weitestgehend nicht mehr verfügbar. In der SAP S/4HANA Simplification List 1909 werden 221 Transaktionen aufgelistet, die in SAP S/4HANA nicht mehr unterstützt werden. Stattdessen kann für bestimmte ausgewählte internationale Handelsfunktionen, wie beispielsweise Intrastat, die Komponente *SAP S/4HANA for International Trade* verwendet werden. Alternativ bietet *SAP Global Trade Services* (SAP GTS) einen umfangreichen Funktionsumfang für die Abwicklung des Außenhandels (z. B. Präferenzmanagement). SAP GTS wird als eigenständiges SAP-System installiert und bietet Standard-schnittstellen für die SAP-S/4HANA-Integration.

Bei einem Umstieg auf SAP S/4HANA sollten Sie prüfen, ob und wie die aktuell verwendeten Außenhandelsprozesse in SAP S/4HANA und/oder SAP GTS abgebildet werden können. Erst dann kann analysiert werden, wie eine Umstellung der Eigenentwicklungen erfolgen kann.

Eingeschränkte
Außenhandels-
funktionalität

Nr. 8: Inkompatible Änderungen des Hauptbuches (SAP-Hinweis 2431747)

Im Finanzwesen wurden in SAP S/4HANA umfangreiche Änderungen durchgeführt, die auch das Hauptbuch betreffen. Aus Entwicklersicht lassen sich die technischen Änderungen wie folgt zusammenfassen:

Umstrukturierung
des Finanzwesens

2

■ Ersetzung von Summen- und Einzelpostentabellen

Viele Summen- und Einzelpostentabellen wurden durch Kompatibilitäts-Views ersetzt (z. B. BSIS, BSAS, BSIK, BSAK, BSID, BSAD, BSIM, MLIT, COEP, COSS, COSP, FAGLBSIS, FAGLBSAS oder FAGLFLEXA). Lesende Zugriffe auf diese Tabellen funktionieren in der Regel noch wie in SAP ERP, sollten aber auch auf ihre Performance hin getestet werden. Schreibende Zugriffe auf diese Kompatibilitäts-Views sind nicht möglich.

■ Speicherung von Belegpositionen in den Tabellen ACDOCA oder BSEG

Die Tabelle BSEG enthält nicht mehr alle Hauptbuchbuchungen. Bestimmte Buchungspositionen z. B. für Umlagen oder Verteilungen aus dem Controlling oder zur Fremdwährungsbewertung werden nur noch in der Tabelle ACDOCA fortgeschrieben, aber nicht mehr in der Tabelle BSEG.

Unabhängig davon, in welcher Tabelle eine Belegposition gespeichert ist, wird der Belegkopf in SAP S/4HANA weiterhin in der Tabelle BKPF fortgeschrieben. Das längerfristige Ziel ist es, die Tabelle BSEG nur noch für die Verwaltung offener Posten oder manueller FI-Buchungen (z. B. über die Transaktion FBO1) zu verwenden.

Über das Kennzeichen BSTAT = 'U' in der Tabelle BKPF können Buchungsvorgänge identifiziert werden, die Einträge in der Tabelle ACDOCA fortzuschreiben, jedoch nicht (mehr) in der Tabelle BSEG sind. Hilfreich bei der Umstellung von Eigenentwicklungen können die beiden Funktionsbausteine FAGL_GET_GL_DOCUMENT und FAGL_GET_LEADING_LEDGER sein. Der Funktionsbaustein FAGL_GET_GL_DOCUMENT wertet die beiden Tabellen ACDOCA und BSEG hinsichtlich der Belegnummer, des Buchungskreises, des Geschäftsjahres und des Ledgers aus. Oftmals wird bei der Selektion das führende Ledger verwendet, das wiederum über den Funktionsbaustein FAGL_GET_LEADING_LEDGER ermittelt werden kann.

■ Eliminierung von Datenbank-Views

Bestimmte Datenbank-Views, wie beispielsweise GLTO_AEDAT, sind nicht mehr vorhanden. Zugriffe auf diese Views müssen umgestellt werden, da es sonst zu Laufzeitfehlern kommt.

■ Neue Customizing-Tabellen

Da in SAP S/4HANA neue Customizing-Tabellen eingeführt wurden, sind bestimmte alte Customizing-Tabellen obsolet bzw. teilweise obsolet. Im

Hauptbuch sind die Tabellen T881, T881T und T882G betroffen. In Eigenentwicklungen sollten Lesezugriffe auf diese Tabellen über die Methoden `CL_FINS_ACDDOC_UTIL=>GET_T881_EMU`, `CL_FINS_ACDDOC_UTIL=>GET_T881T_EMU` und `CL_FINS_ACDDOC_UTIL=>GET_T882G_EMU` umgestellt werden.



Architektur des Hauptbuches im Überblick

Die in SAP S/4HANA durchgeführten Änderungen am Hauptbuch sind sehr umfangreich. Eine detaillierte Auflistung aller Änderungen würde den Rahmen dieses Abschnitts sprengen. Deshalb verweisen wir an dieser Stelle auf den SAP-Hinweis 2431747, der detailliertere Informationen zur Umstellung liefert. Die Anlage zu diesem Hinweis veranschaulicht die neue Architektur mit grafischen Beispielen und liefert Anleitung zur Anpassung des kundeneigenen Codes.

Nr. 9: Vertriebsunterstützung (SAP-Hinweis 2214585)

Die Vertriebsunterstützung (Computer-Aided-Selling, kurz CAS) wird in SAP S/4HANA nicht mehr unterstützt. Die strategische Ausrichtung von SAP sieht vor, dass entsprechende Funktionen zukünftig durch *SAP Sales Cloud* im Portfolio *SAP Customer Experience* (vormals SAP C/4HANA) oder in *SAP Customer Relationship Management* (SAP CRM) abgebildet werden sollen.

Blacklist-Monitor

In der SAP S/4HANA Simplification List 1909 werden 55 Transaktionen aufgelistet, die nicht mehr unterstützt werden. Diese Transaktionen wurden in einen *Black List Monitor* aufgenommen. Bei der Verwendung von Entwicklungsobjekten wie Transaktionen, Funktionsbausteinen, Reports usw., die in die Ausschlussliste des Black List Monitors aufgenommen wurden, wird bei einer Verwendung automatisch ein Fehler ausgelöst. Dort können sie auch nach einer nachträglichen Modifikation der Ausschlusslisteneinträge im Black List Monitor in der Regel nicht mehr aufgerufen werden, da viele abhängige Objekte in SAP S/4HANA nicht mehr verfügbar sind. SAP-Hinweis 2249880 gibt einen Überblick zum Vorgehen bei Entwicklungsobjekten, die von SAP in den Black List Monitor aufgenommen wurden.

Ob und in welchem Umfang das kundeneigene Code bei Befunden zur Vertriebsunterstützung angepasst werden muss, ist abhängig davon, welche Prozesse Sie aktuell und zukünftig verwenden wollen. Im Folgenden geben wir Ihnen deshalb am Beispiel der Vereinfachungen in der Komponente Vertriebsunterstützung einen allgemeinen Überblick, wie mit obsoleten Zugriffen auf Entwicklungsobjekte umgegangen werden kann, die von SAP in den Black List Monitor aufgenommen wurden:

■ Technische Verwendung von Objekten der Vertriebsunterstützung

Der Custom Code Check prüft die Eigenentwicklung auf mögliche Inkompatibilitäten in SAP S/4HANA. Nicht Bestandteil dieser Prüfung ist, ob und wie die vom Coding abhängigen Standardprozesse operativ verwendet werden. Häufig greifen Entwickler auf Entwicklungsobjekte wie Datenelemente, Strukturen oder Datenbanktabellen zurück, weil sie technisch passend scheinen, ohne Kenntnis von den abhängigen SAP-Standardprozessen zu haben.

Bei einer solchen *technischen Verwendung* ist es in der Regel ausreichend, das auf der Ausschlussliste stehende Vertriebsunterstützungsobjekt durch ein anderes freigegebenes SAP-Objekt oder Z-Objekt zu ersetzen. In der Regel ist die Verwendung freigegebener SAP-Objekte der Neuanlage kundeneigener Objekte vorzuziehen, z. B. weil Übersetzungen bereits vorhanden sind, der Wartungsaufwand reduziert werden kann usw.

■ Fachliche Verwendung von Objekten der Vertriebsunterstützung

Aufwendiger sind die Analyse und Umsetzung bei einer fachlichen Verwendung der SAP-Vertriebsunterstützungsobjekte. Wenn beispielsweise in SAP ERP die Transaktion V+01 verwendet wird, um einen Kontakt anzulegen, wird dieser Prozessschritt in SAP S/4HANA nicht mehr funktionieren. Die Transaktion V+01 ist Bestandteil der Ausschlussliste des Black List Monitors und wird in SAP S/4HANA nicht mehr unterstützt. Auch wenn bestimmte Strukturen oder Datenbanktabellen zur Vertriebsunterstützung noch existieren, sollten diese nicht mehr verwendet werden, da die übergeordneten Geschäftsprozesse teilweise oder vollständig nicht mehr ausführbar sind.

Bevor das Entwicklungsobjekt angepasst werden kann ist eine konzeptionelle Aufarbeitung notwendig, wie zukünftig der Prozess zur Vertriebsunterstützung mit SAP S/4HANA abgebildet werden soll. Möglicherweise wird eine Eigenentwicklung mit der Einführung von SAP Customer Experience oder SAP CRM obsolet, da dort bestimmte Funktionalitäten bereits im SAP-Standard zur Verfügung stehen. Eine Bewertung der Auswirkungen auf die Eigenentwicklungen ist erst nach Spezifizierung des Soll-Prozesses mit SAP S/4HANA möglich.

Analyse der Prozesse

Nr. 10: Indirekte Beziehungen in der Vertriebsbelegflusstabelle VBFA (SAP-Hinweis 2470721)

In der Datenbanktabelle VBFA werden die Verknüpfungen zu bestimmten SD-Belegen, wie beispielsweise Kundenaufträgen, Lieferungen, Ausgangsrechnungen oder FI-Belegen, als *Vertriebsbelegfluss* gespeichert. Über die

Felder VBELV/POSNV und VBELN/POSNN in der Tabelle VBFA wird jeweils der Vorgängerbeleg mit dem Nachfolgebeleg verknüpft. In SAP ERP wurden auch indirekte Beziehungen zu den Belegen in der Tabelle VBFA fortgeschrieben. Das Feld STUFE in der Tabelle VBFA beschreibt dabei die *relative Distanz* der Belege zueinander. Die relative Distanz legt somit die direkte oder indirekte Belegbeziehung vom Vorgängerbeleg zum Nachfolgebeleg fest.

Direkte vs. indirekte Beziehung

Den Unterschied zwischen direkter und indirekter Belegbeziehung möchten wir Ihnen an zwei Beispielen verdeutlichen:

- **Kundenauftrag (Vorgänger) und Auslieferung (Nachfolger)**
Das Feld STUFE hat den Wert »00«, da die Auslieferung in einer *direkten Beziehung* zum Kundenauftrag angelegt wurde.
- **Kundenauftrag (Vorgänger) und Rechnung (Nachfolger)**
Das Feld STUFE hat den Wert »01«, da die Rechnung in einer *indirekten Beziehung* zum Kundenauftrag steht und exakt *ein* weiterer Beleg, die Auslieferung, zwischen dem Kundenauftrag und der Rechnung verbucht wurde.

Re-Einführung von STUFE

In früheren SAP-S/4HANA-Versionen wurde das Feld STUFE aus der Tabelle VBFA entfernt. Damit war nur noch eine Fortschreibung von direkten Beziehungen möglich. Ab den im Folgenden genannten Releases und Support-Package-Ständen wurden das Feld STUFE und die Speicherung von indirekten Beziehungen wieder eingeführt:

- SAP S/4HANA 1709 (SPO0)
- SAP S/4HANA 1610 SPO2 und FPS2 (Feature-Pack-Stack 2)
- SAP S/4HANA 1511 SPO4
- über die Korrekturanleitung des SAP-Hinweises 2418242 in SAP S/4HANA 1610 SPO0/SPO1
- über die Korrekturanleitung des SAP-Hinweises 2418242 in SAP S/4HANA 1511 SPO0/SPO1/SPO2/SPO3

Bei einer Systemkonvertierung ausgehend von einem Release ohne das Feld STUFE ist ein Neuaufbau der fehlenden Belegflusseinträge für die indirekten Beziehungen notwendig.

Die Umstellung der Eigenentwicklungen ist somit stark vom Zielrelease abhängig. Zum aktuellen Zeitpunkt ist das Ausgangssystem bei einer Systemkonvertierung in den meisten Fällen ein SAP-ERP-System, und dieses wird in ein aktuelles SAP-S/4HANA-Release konvertiert. Da in diesen Fällen beide Produkte die indirekten Beziehungen in der Tabelle VBFA verwenden, kann die Änderung hinsichtlich des Feldes STUFE bei der Anpassung

der Eigenentwicklungen vernachlässigt werden. Trotz der Ablösung des Primärschlüssels aus SAP ERP mit Vorgängerbelegnummer, Vorgängerbelegposition, Nachfolgebelegnummer, Nachfolgebelegposition und Vertriebsbelegtyp des Nachfolgebelegs durch die eindeutige Identifikation der Belegbeziehung (RUUID) in SAP S/4HANA können Zugriffe über den alten Primärschlüssel in SAP S/4HANA weitestgehend bestehen bleiben.

Vorwort

»Hallo Basisentwicklung, ihr habt doch da so einen Verwendungsnachweis in ABAP. Damit könnte man die Kunden doch unterstützen, ihren Z-Code anzupassen, wenn sie ihr System auf SAP S/4HANA umstellen, oder?«

Diese Anfrage war der Startschuss für das Projekt *Custom-Code-Migration*, das wir im März 2015 mit dem Ziel begonnen haben, unsere Kunden bestmöglich dabei zu unterstützen, ihr System von der klassischen SAP Business Suite auf SAP S/4HANA umzustellen. Was aus dieser Anfrage geworden ist, können Sie in diesem Buch erfahren.

Heute, etwa fünf Jahre später, ist das Thema Custom-Code-Migration bei der Mehrzahl der SAP-Kunden eines der brennendsten Themen in den IT-Abteilungen. Denn ihre Systemlandschaften befinden sich gerade im Umbruch in Richtung SAP S/4HANA. Dabei muss sehr gut geplant werden, was mit den ganzen Eigenentwicklungen der letzten Jahrzehnte passieren soll. In diesem Kontext treten häufig ähnliche Fragen wie die folgenden auf:

- Muss ich meinen gesamten Kundencode nochmals von Neuem entwickeln?
- Darf ich in SAP S/4HANA überhaupt noch Z-Code entwickeln?
- Wie sieht es mit meinen Modifikationen und Erweiterungen aus?

Was ist hier eigentlich das Problem, das gelöst werden muss? Das S in S/4HANA steht ja für Simplifizierung. Das klingt erstmal positiv. Es scheint alles einfacher zu werden. Aber für den Entwickler von kundeneigenem Code bedeutet gerade diese Simplifizierung auch viel Arbeit. Warum?

Simplifizierung bedeutet, dass jede Menge Objekte im SAP-Code geändert wurden. Einige Objekte wurden inkompatibel geändert oder sogar gelöscht. Das hat zum Teil gravierende Auswirkungen auf den kundeneigenen Code, etwa Abbrüche zur Laufzeit, Syntaxfehler in den Programmen oder im schlimmsten Fall funktionale Fehler in den Anwendungen, die teilweise nur schwer auffindbar sind. Aus diesem Grund muss in jedem SAP-S/4HANA-Projekt der Aufwand geleistet werden, den kundeneigenen Code zu analysieren und entsprechend anzupassen.

Zum Glück hat sich gezeigt, dass diese Aufgabe mittlerweile sehr gut machbar ist. Sie ist auch deshalb gut machbar, weil es inzwischen Experten gibt, die sehr viel Projekterfahrung auf diesem Gebiet gesammelt haben. Zwei dieser Experten haben sich entschieden, ihre Erfahrungen in diesem Buch

zusammenzustellen, um Ihnen zu helfen, Ihre Codeumstellungen zu meistern. Es bedarf nämlich nicht nur allein der Kenntnis der richtigen Werkzeuge und der in SAP S/4HANA vorgenommenen Änderungen im Vergleich zur SAP Business Suite, sondern es muss auch die richtige Vorgehensweise gefunden werden, um so ein Projekt effizient zu meistern.

Schon die ersten Kundenprojekte mit dem Release 1511 haben gezeigt, dass eine reine Anpassung des existierenden Kundencodes oftmals nicht ausreicht, um sich erfolgreich auf die neue Business Suite vorzubereiten. Eine ebenso wichtige Aufgabe ist es, sich zu überlegen, ob denn wirklich jede Zeile ABAP-Code, die in den letzten Jahren geschrieben wurde, in SAP S/4HANA überhaupt noch benötigt wird. Ich ziehe hier immer gern den Vergleich zum Umzug in ein neues Haus oder eine neue Wohnung. Auch dabei ist es sehr ratsam, zuvor zu überlegen, ob man wirklich all seine Sachen mitnehmen muss oder ob man sich nicht von dem ein oder anderen Ding trennen kann.

Genau so sieht es aus, wenn man nach SAP S/4HANA umzieht. Es gibt Analysen, die zeigen, dass ca. 40–60 % des gesamten Kundencodes nicht produktiv genutzt werden. Warum also sollte man diesen Code in dem Wissen anpassen, dass er auch in SAP S/4HANA nicht mehr genutzt werden wird? Besser ist es, diesen Code vor der Systemumstellung zu löschen. Dafür bietet SAP Tools an, die Sie anleiten, um den überflüssigen Code zu identifizieren und zu eliminieren.

Wenn man diesen ersten Schritt gemeistert hat, geht es an die eigentliche Analyse des noch immer relevanten Kundencodes, der mit nach SAP S/4HANA umziehen soll. Wir bieten dafür mit der Custom-Code-Migration-App, basierend auf dem ABAP Test Cockpit, nicht nur ein sehr effizientes Analysewerkzeug an, sondern mit den sogenannten Quick Fixes mittlerweile auch Ansätze, um den Kundencode semiautomatisch umzusetzen. Mit diesen Werkzeugen sollte jeder Kunde in der Lage sein, die notwendigen Anpassungen vorzunehmen.

Der nächste Schritt, der aktuell in vielen Projekten noch nicht angegangen wird, ist die Modernisierung und Optimierung der kundeneigenen Entwicklungen. Sie kaufen sich mit SAP S/4HANA ja nicht nur eine modernisierte und vereinfachte Anwendungsschicht, sondern auch eine moderne und renovierte ABAP-Technologieplattform, basierend auf SAP HANA. Auch in Ihren eigenen Anwendungen können Sie Nutzen aus diesen neuen Technologien ziehen. Gerade die SAP-HANA-Technologie bietet Ihnen als Entwickler sehr viele Möglichkeiten, die Performance Ihrer Anwendungen deutlich zu verbessern, indem datenintensive Operationen direkt auf der Datenbank ausgeführt werden und nicht wie früher im ABAP-Stack. Das setzt aber voraus, dass Sie sich mit Technologien wie Core Data Services

und Datenbankprozeduren auseinandersetzen. Natürlich fällt dieses Wissen nicht vom Himmel, sondern ist mit entsprechenden Investitionen und Aufwänden verbunden, um diese neuen Technologien im Rahmen eines SAP-S/4HANA-Projekts zu erlernen. Viele Kunden stellen erst mitten im Projekt fest, dass sie in SAP S/4HANA einen CDS View anlegen oder erweitern müssen und dass dafür eine neue Entwicklungsumgebung zu verwenden ist. Viele dieser Technologien existieren jedoch schon seit geraumer Zeit. Sie müssen also gar nicht Ihr SAP-S/4HANA-Projekt abwarten, um sie einzusetzen.

Zusammenfassend lautet mein Vorschlag für einen erfolgreichen Umgang mit dem Kundencode in einem SAP-S/4HANA-Projekt:

1. Identifizieren und eliminieren Sie nicht genutzten Code. Denken Sie an den Umzug ins neue Haus!
2. Nutzen Sie die Werkzeuge, die SAP bereitstellt, um Ihren Kundencode zu analysieren und anzupassen.
3. Nutzen Sie die Chance, Ihre Anwendungen mithilfe der neuen Technologien zu optimieren und zu vereinfachen.

Wenn ich Sie überzeugen konnte, diesen Weg einzuschlagen, zögern Sie nicht, weiterzulesen, um alle Details zu dieser Vorgehensweise zu erfahren.

Thomas Fiedler

SAP, Product Owner ABAP Development Tools

Einleitung

Für viele SAP-Kunden steht in den nächsten Jahren die Umstellung ihrer SAP Business Suite auf SAP S/4HANA an. Durch die für SAP S/4HANA charakteristischen Vereinfachungen im Vergleich zur klassischen SAP Business Suite ist dieser Umstieg häufig sehr anspruchsvoll. Er bietet allerdings auch großes Potenzial, die eigenen Prozesse zu optimieren und von zahlreichen Innovationen zu profitieren. Doch ist es insbesondere für SAP-Kunden mit zahlreichen und komplexen Eigenentwicklungen in den Systemen entscheidend, diese Prozesse und Funktionen auch in der neuen SAP-S/4HANA-Welt weiterhin nutzen zu können. Dabei steht zum einen die Sicherung getätigter Investitionen und zum anderen die Umsetzung der eigenen Prozesse außerhalb von SAP-Standardlösungen oder Partner-Add-ons im Zentrum des Interesses.

Zur Unterstützung hat SAP Werkzeuge und Vorgehensweisen entwickelt, um die kundeneigenen Entwicklungen, den sogenannten *Custom Code*, nach SAP S/4HANA zu migrieren. Der Begriff *Custom-Code-Migration* wird dabei als Oberbegriff für die Übernahme und Anpassung der Eigenentwicklung verwendet.

Custom-Code-Migration

Das Ziel dieses Buches ist es, Ihnen Wege und Lösungen vorzustellen, wie Sie Ihre Eigenentwicklungen in SAP S/4HANA möglichst kostengünstig und zielgerichtet übernehmen können. Zur besseren Veranschaulichung finden Sie dazu zahlreiche praktische Beispiele und Empfehlungen auf Basis unserer umfangreichen Projekterfahrungen.

Ziel dieses Buches

Da dieses Buch einen kompletten Überblick über die Migration der Eigenentwicklungen nach SAP S/4HANA gibt, ist es für unterschiedliche Lesergruppen geeignet. Zur Vorbereitung des Umstiegs auf SAP S/4HANA und für eine Bewertung der Auswirkungen dieses Umstiegs auf die Eigenentwicklungen richtet es sich an IT-Manager und -Mitarbeiter, die in absehbarer Zeit den Umstieg auf SAP S/4HANA planen. Darüber hinaus gibt es Projektleitern oder Entwicklungsmanagern einen Überblick über die einzelnen Werkzeuge und die Methodik, damit sich diese innerhalb des Angebots orientieren und dessen Relevanz für das eigene Transformationsprojekt bewerten können. Für ABAP-Entwickler wird eine Vielzahl praxisnaher Beispiele vorgestellt, um die Umstellung der Eigenentwicklungen im Detail zu erläutern.

Zielgruppe

Für die Lektüre dieses Buches sind Kenntnisse der ABAP-Entwicklung und ein gewisses Grundverständnis von SAP NetWeaver und SAP ERP von Vorteil. Weiterführende Informationen zur ABAP-Entwicklung, SAP HANA, SAP S/4HANA oder SAP NetWeaver finden Sie in spezialisierten Büchern.

Vorkenntnisse

SAP-S/4HANA-
Ausgangsrelease

Dieses Buch beschreibt die Vorgehensweise für die Migration der Eigenentwicklung in SAP-S/4HANA-Transformationsprojekten auf Basis des aktuellen Releases SAP S/4HANA 1909. Bei höheren Releases kann es deshalb zu Abweichungen und anderen Bildschirmansichten im Vergleich zu den in diesem Buch vorgestellten Abbildungen kommen. Aktuell ist jedoch davon auszugehen, dass die generellen Schritte zur Durchführung der Migration der Eigenentwicklungen auch in höheren SAP-S/4HANA-Releases weitestgehend gleich bleiben werden.

Aufbau dieses
Buches

Die ersten Kapitel sind gleichermaßen für IT-Manager und IT-Mitarbeiter interessant, die noch kein konkretes Transformationsprojekt geplant haben und sich über verschiedene Möglichkeiten informieren möchten.

In **Kapitel 1**, »Was kommt mit SAP S/4HANA auf Sie zu?«, führen wir Sie in die Neuerungen in SAP S/4HANA im Vergleich zur klassischen SAP Business Suite ein. Anschließend erhalten Sie einen Überblick der drei grundlegenden Transformationsszenarien Neuimplementierung, Systemkonvertierung und Landschaftstransformation. Darauf aufbauend werden die Unterschiede bei der Übernahme von kundeneigenen Entwicklungen in den jeweiligen Transformationsszenarien erläutert.

In **Kapitel 2**, »Das Konvertierungsprojekt im Überblick«, werden anhand eines Beispiels der typische Ablauf eines Konvertierungsprojekts und die notwendigen Tätigkeiten für die Migration der Eigenentwicklungen erörtert. Anschließend werden hilfreiche vorbereitende Maßnahmen für den Umstieg nach SAP S/4HANA vorgestellt.

Die folgenden Kapitel geben IT-Managern, Administratoren und Entwicklern einen Überblick der notwendigen Anpassungen an Eigenentwicklungen.

In **Kapitel 3**, »Umfang der Custom-Code-Migration bestimmen«, stellen wir Ihnen verschiedene Lösungen zur Ermittlung der produktiven Eigenentwicklungen vor. In vielen Fällen wird es Altlasten in Form von Eigenentwicklungen geben, die nicht mehr genutzt werden. Die Reduzierung des Migrationsumfangs um diese nicht mehr benötigten Eigenentwicklungen verringert wesentlich den Aufwand für die Umstellung auf SAP S/4HANA.

In **Kapitel 4**, »Analyse der erforderlichen Anpassungen«, zeigen wir Ihnen, wie Sie Ihre Entwicklungen auf die Kompatibilität mit SAP HANA oder SAP S/4HANA hin prüfen zu können. Hierfür erörtern wir die Analysewerkzeuge von SAP, wie beispielsweise das ABAP Test Cockpit, den Code Inspector oder die App *Custom Code Migration*.

Die folgenden Kapitel richten sich gezielt an ABAP-Entwickler und fokussieren sich auf die Codeumstellung für SAP S/4HANA.

In **Kapitel 5**, »Custom-Code-Anpassung für SAP HANA«, wird die Umstellung von datenbankspezifischen Eigenentwicklungen im Kontext von SAP HANA anhand von vielen praktischen Beispielen vorgestellt. Hierbei werden nicht nur performancerelevante Auswirkungen betrachtet, sondern auch potenzielle Fehler, die innerhalb der Eigenentwicklung auf SAP HANA auftreten können.

In **Kapitel 6**, »Custom-Code-Anpassung für SAP S/4HANA«, werden alle SAP-S/4HANA-Prüfvarianten vorgestellt, und die Umstellung der Eigenentwicklung für SAP S/4HANA wird auf Basis vieler praktischer Beispiele erläutert. Dabei zeigen wir Ihnen konkrete Lösungsvorschläge zu allen gängigen Prüfmeldungen auf.

Zur Reduzierung des manuellen Aufwands der Custom-Code-Migration stellt SAP *Quick Fixes* zur Verfügung, die eine halbautomatische Codekorrektur auf Basis von Best Practices erlauben. In **Kapitel 7**, »Quick Fixes«, erhalten Sie deshalb einen Überblick zu deren Funktionsumfang und ihrer Verwendung.

In **Kapitel 8**, »Custom-Code-Optimierung nach der Migration«, betrachten wir das Optimierungspotenzial von Eigenentwicklungen. Wir erläutern, wie Sie die kritischen Teile der zu optimierenden Eigenentwicklungen, z. B. mit dem SQL-Monitor, identifizieren können. Zusätzlich gehen wir auf einige Neuerungen in der ABAP-Sprache und die neuen Erweiterungstechnologien für SAP S/4HANA ein.

In **Kapitel 9**, »Best Practices für Transformationsprojekte«, erörtern wir die allgemeinen Herausforderungen in Transformationsprojekten und geben Ihnen Empfehlungen. Abschließend stellen wir vier am Markt etablierte Partnertools und -services vor, die den SAP-S/4HANA-Umstieg mit speziellen Lösungen in unterschiedlichen Bereichen unterstützen oder ihn erleichtern können.

In hervorgehobenen Informationskästen finden Sie in diesem Buch Inhalte, 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:

- In Kästen, die mit diesem Symbol gekennzeichnet sind, finden Sie Informationen zu *weiterführenden Themen* oder wichtigen Inhalten, die Sie sich merken sollten.
- Die mit dem *Tippsymbol* gekennzeichneten Kästen geben Ihnen spezielle Empfehlungen, die Ihnen die Arbeit erleichtern können.
- Kästen mit dem *Achtungssymbol* weisen Sie auf typische Probleme oder Fallstricke hin.

Informationskästen





- Durch dieses Symbol kenntlich gemachte *Beispiele* weisen auf Einsatzbeispiele aus der Praxis hin.

SAP hat mit dem Übergang zu SAP S/4HANA und den umgebenden Cloud-Lösungen eine starke technologische Innovation umgesetzt, die mit dem Wechsel von R/2 nach R/3 vergleichbar ist. In diesem Buch beschreiben wir die vielen Herausforderungen und Lösungsansätze, die so ein Umbruch mit sich bringt. Lassen Sie sich auch von dem Gedanken leiten, dass der Wechsel in die neue SAP-S/4HANA-Welt große Chancen und Möglichkeiten bietet. Daran mitzuwirken ist eine spannende Erfahrung für jeden Beteiligten und bereichernd für das betreffende Unternehmen.

Wir hoffen, dass Sie dieses Buch als hilfreiches Nachschlagewerk und Leitfaden für Ihre Reise nach SAP S/4HANA wahrnehmen. Wir wünschen Ihnen viel Vergnügen beim Lesen!

Danksagung

Wir bedanken uns für die freundliche Unterstützung und das wertvolle, konstruktiv-kritische Feedback bei Christian Kuhn, Annett Laube, Marco Wahler und Lars Woll. Weiterhin gilt unser herzlicher Dank den Kollegen bei SAP für die großzügige Begleitung des Projekts, insbesondere namentlich und stellvertretend Thomas Fiedler. Abschließend danken wir Janina Schweitzer von SAP PRESS für ihre kompetente Unterstützung und die sehr gute Zusammenarbeit bei der Erstellung dieses Buches.

Emily Celen und Lutz Rosenpflanze

Auf einen Blick

1	Was kommt mit SAP S/4HANA auf Sie zu?	23
2	Das Konvertierungsprojekt im Überblick	45
3	Umfang der Custom-Code-Migration bestimmen	99
4	Analyse der erforderlichen Anpassungen	141
5	Custom-Code-Anpassung für SAP HANA	195
6	Custom-Code-Anpassung für SAP S/4HANA	247
7	Quick Fixes	311
8	Custom-Code-Optimierung nach der Migration	331
9	Best Practices für Transformationsprojekte	385

Inhalt

Vorwort	15
Einleitung	19

1 Was kommt mit SAP S/4HANA auf Sie zu? 23

1.1 SAP S/4HANA im Vergleich zur klassischen SAP Business Suite	25
1.1.1 Der neue Geschäftspartner	26
1.1.2 Das Universal Journal	27
1.1.3 SAP Credit Management	28
1.1.4 Embedded Analytics	28
1.1.5 Die neue Materialbedarfsplanung	29
1.1.6 Die neue Produktions- und Feinplanung	30
1.1.7 SAP Fiori	30
1.1.8 Das neue Erweiterungskonzept	32
1.1.9 Intelligente Technologien	33
1.1.10 SAP S/4HANA Cloud	34
1.2 Transformationsszenarien	34
1.2.1 Custom-Code-Migration im Rahmen einer Neuimplementierung	37
1.2.2 Custom-Code-Migration im Rahmen einer Systemkonvertierung	40
1.2.3 Custom-Code-Migration im Rahmen einer Landschaftstransformation	43
1.3 Zusammenfassung	44

2 Das Konvertierungsprojekt im Überblick 45

2.1 Ablauf eines Konvertierungsprojekts	45
2.2 Beispiel für einen groben Projektplan	51

2.3 Welche Änderungen am Anwendungscode sind erforderlich?	54
2.3.1 Die Vereinfachungsdatenbank	54
2.3.2 Kompatibilitäts-Views	56
2.3.3 Die Top 10 der Simplification Items	64
2.4 Die wichtigsten Vorbereitungen für die Konvertierung	83
2.4.1 SAP Readiness Check	85
2.4.2 SAP Business Scenario Recommendations	86
2.4.3 SAP Fiori Apps Recommendations	87
2.4.4 Maintenance Planner	88
2.4.5 SAP Activate	89
2.4.6 SAP Transformation Navigator	90
2.4.7 Aktuelle Systemlandschaftskarte und Schnittstellen-übersicht	91
2.4.8 Stilllegung nicht mehr benötigter Eigenentwicklungen ...	92
2.4.9 Custom Code Check: Anpassungen in SAP ERP	94
2.4.10 Systemkopien von Produktiv- und Qualitätssicherungssystem	94
2.4.11 Simplification Item Check	95
2.4.12 Datenbereinigung und Datenarchivierung	96
2.5 Zusammenfassung	97

3 Umfang der Custom-Code-Migration bestimmen 99

3.1 Welcher Code muss migriert werden?	99
3.1.1 Welche Objekte werden nach der Migration nicht mehr benötigt?	100
3.1.2 Welche Objekte werden bereits nicht mehr verwendet?	101
3.2 Verwendungsprotokollierung	102
3.2.1 Usage and Procedure Logging	102
3.2.2 ABAP Call Monitor	106
3.2.3 SQL-Monitor	112
3.2.4 Coverage Analyzer	113
3.2.5 Eigene Tools zur Verwendungsprotokollierung	115

3.3 Custom Code Lifecycle Management im SAP Solution Manager	116
3.3.1 Analyse anlegen	117
3.3.2 Objektliste auswerten	120
3.3.3 Ausschlussliste verwenden	121
3.3.4 Vormerkung zur Löschung	124
3.4 Die App »Custom Code Migration«	124
3.4.1 Aktivieren der App auf der SAP Cloud Platform	125
3.4.2 Verwendungsprotokollierung auswerten	126
3.5 Manuelle Auswertung der Verwendung kundeneigener Objekte	128
3.5.1 Priorisierung der zu löschenden Objekte	129
3.5.2 Bestimmung indirekter Verwendungen	131
3.5.3 Archivierung einer Kopie der zu löschenden Objekte	134
3.5.4 Löschen von Objekten	137
3.6 Zusammenfassung	138

4 Analyse der erforderlichen Anpassungen 141

4.1 Welche Tools helfen bei der Analyse?	141
4.2 ABAP Test Cockpit	144
4.2.1 Vorbereitungen zur Nutzung des ABAP Test Cockpits	144
4.2.2 ATC-Prüfsystem einrichten	146
4.2.3 Prüfläufe einplanen	148
4.2.4 Darstellung der Befunde im ATC-Ergebnis-Browser	150
4.2.5 Steuerung und Darstellung der Prüfergebnisse im SAP-ERP-System	154
4.3 Standardprüfvarianten	156
4.3.1 SAP-HANA-Datenbank-Prüfungen	156
4.3.2 SAP-S/4HANA-Prüfungen	159
4.3.3 Default-Prüfvarianten	166
4.4 Prüfung fremder Namensräume	168
4.5 Prüfungen von SAP-Fiori-Apps aus SAP ERP	170
4.6 Prüfungen mit der App »Custom Code Migration«	173
4.6.1 Aktivieren der App in SAP S/4HANA on premise	174
4.6.2 Migrationsprojekt einrichten	175

4.6.3	Analyse und Darstellung der Befunde	176
4.6.4	Transportauftrag erstellen	180
4.6.5	Ausblick: Geplante Neuerungen	181
4.7	Dateibasierte Werkzeuge	182
4.7.1	Custom Code Migration Worklist	183
4.7.2	Extraktbasierte Prüfungen im Code Inspector	185
4.8	Weitere Prüfungen und Aktivitäten im Rahmen der Systemkonvertierung	188
4.9	Zusammenfassung	193
5	Custom-Code-Anpassung für SAP HANA	195
5.1	Technische Grundlagen zur Custom-Code-Anpassung für SAP HANA	196
5.1.1	Implizite Sortierung	198
5.1.2	Natives SQL	199
5.1.3	Datenbank-Hints	200
5.1.4	Pool- und Cluster-Tabellen	202
5.1.5	Prüffehler	204
5.2	Beispiele für Prüffehler	206
5.2.1	Prüfung kann wegen fehlender Voraussetzungen nicht durchgeführt werden	206
5.2.2	Sonstige Prüffehler	207
5.3	Beispiele für den Prüftitel »Voraussetzungen für den Test«	209
5.3.1	Verwendetes Objekt existiert nicht	209
5.3.2	Syntaxfehler	210
5.4	Beispiele für den Prüftitel »Kritische Anweisungen«	212
5.4.1	Verwendung von nativen SQL-Anweisungen	212
5.4.2	Verwendung eines Datenbank-Hints	214
5.5	Beispiele für den Prüftitel »Problematische Anweisungen«	215
5.5.1	SELECT ... FOR-Cluster-Tabelle ohne ORDER BY	216
5.5.2	DELETE ADJACENT DUPLICATES	217
5.5.3	LOOP AT itab AT ... ENDAT	219
5.5.4	Leere SELECT ... ENDSELECT-Anweisung	220
5.5.5	READ ... BINARY SEARCH	222
5.5.6	READ TABLE ... INDEX 1	225
5.5.7	SELECT SINGLE möglicherweise nicht eindeutig	226

5.6	Beispiele für den Prüftitel »DB-Operationen in Pool-/Cluster-Tabellen suchen«	229
5.6.1	Datenbankoperation SELECT für Tabellen-Cluster RFBLG	229
5.6.2	Datenbankoperation SELECT für Tabellen-Pool UTAB	232
5.7	Beispiele für den Prüftitel »Verwendung der ADBC-Schnittstelle«	235
5.7.1	Verwendung der ADBC-Klasse CL_SQL_STATEMENT	235
5.7.2	Verwendung der ADBC-Klasse CL_SQL_CONNECTION	237
5.8	Suche von ABAP-Anweisungsmustern	239
5.8.1	Aufruf des Funktionsbausteins DB_EXISTS_INDEX	240
5.8.2	Aufruf des Funktionsbausteins DD_INDEX_NAME	244
5.9	Zusammenfassung	246
6	Custom-Code-Anpassung für SAP S/4HANA	247
6.1	Technische Grundlagen zur Custom-Code-Anpassung für SAP S/4HANA	248
6.1.1	Direkte Tabellenzugriffe	249
6.1.2	Literale im Code	250
6.1.3	ABAP-Dictionary-Objekte zu abgekündigten Lösungen	252
6.1.4	Zeitpunkt der Codeanpassungen	253
6.1.5	Intention und Ziel von Eigenentwicklungen	254
6.1.6	Testen von APIs	255
6.2	Versionen der SAP-S/4HANA-Prüfvarianten	257
6.3	Beispiele für den Prüftitel »S/4HANA: Feldlängen- erweiterungen«	258
6.3.1	Vergleich-Längenkonflikt (Domäne MATNR)	259
6.3.2	Strukturkomponenten-Typkonflikt (Domäne MATNR)	262
6.3.3	Alter Konkatenieren-Längenkonflikt (Domäne MATNR)	264
6.3.4	CALL FUNCTION: Generischer Parameter (Domäne MATNR)	267
6.3.5	Typkonflikt (Datenelement FWSTEV)	270
6.4	Beispiele für den Prüftitel »S/4HANA: Nach ABAP-Dictionary- Erweiterungen suchen«	274
6.4.1	Append erweitert ein vereinfachtes Artefakt (Tabelle MARC)	274

6.4.2	Append erweitert ein vereinfachtes Artefakt (Tabelle COEP)	279
6.5	Beispiele für den Prüftitel »S/4HANA: Nach Basistabellen der ABAP-Dictionary-Views suchen«	281
6.5.1	View, basierend auf einem ersetzten Artefakt (Tabelle KONV)	282
6.5.2	View, basierend auf einem Artefakt mit einem Ersetzungsobjekt (Tabelle MKPF)	284
6.6	Beispiele für den Prüftitel »S/4HANA: Suche nach Datenbankoperationen«	288
6.6.1	DB-Operation UPDATE wurde gefunden (Tabelle LFA1) ...	288
6.6.2	DB-Operation SELECT wurde gefunden (Tabelle KONV) ...	291
6.7	Beispiele für den Prüftitel »S/4HANA: Suche nach Verwendungen der vereinfachten Objekte«	293
6.7.1	Syntaktisch nicht kompatible Änderung der vorhandenen Funktion (Datenelement VB Typ)	293
6.7.2	Semantisch nicht kompatible Änderung der vorhandenen Funktion (Transaktion XD02)	295
6.7.3	Verwendung eines nicht mehr verfügbaren Entwicklungsobjekts (Funktionsbaustein GET_WEEK_INFO_BASED_ON_DATE)	297
6.7.4	Funktion nicht verfügbar: funktionales Äquivalent verfügbar (Datenelement /BEV3/CHOBJBEZ)	300
6.7.5	Funktion nicht verfügbar: gleichwertige Funktion auf Roadmap (Datenelement CHAR02)	302
6.8	Beispiele für den Prüftitel »S/4HANA: Readiness Check für SAP Queries«	305
6.8.1	InfoSet verwendet simplifizierte Tabellen (Tabelle KONV)	306
6.8.2	InfoSet inkonsistent	308
6.9	Zusammenfassung	309
7	Quick Fixes	311
7.1	Systemvoraussetzungen	311
7.2	Funktionsumfang der SAP-S/4HANA-Quick-Fixes	313
7.2.1	Quick Fixes für verschiedene Anwendungsfälle	314

7.2.2	Übersicht in der App »Custom Code Migration«	315
7.2.3	Automatische Generierung von Kommentaren	316
7.3	Quick Fixes in den ABAP Development Tools verwenden	316
7.3.1	Installation und Konfiguration der ABAP Development Tools	316
7.3.2	Objektprüfung durchführen	318
7.3.3	Quick Fixes anwenden	320
7.4	Eigene Quick Fixes anlegen	325
7.5	Zusammenfassung	329
8	Custom-Code-Optimierung nach der Migration	331
8.1	Optimierungswerkzeuge	333
8.1.1	SQL-Monitor	334
8.1.2	Arbeitsvorrat für die SQL-Performanceoptimierung	337
8.1.3	ATC-Checks zur Performanceoptimierung für SAP HANA	341
8.1.4	Laufzeitprüfungsmonitor	342
8.2	Optimierung der Datenbankzugriffe	344
8.2.1	Code-Pushdown-Prinzip	344
8.2.2	Core Data Services	347
8.2.3	ABAP Managed Database Procedures	353
8.2.4	Die 5 goldenen Regeln	355
8.3	Neue Sprachelemente und Programmierobjekte	356
8.3.1	Wichtige Neuerungen in Open SQL	357
8.3.2	Wichtige Neuerungen in ABAP	359
8.3.3	Metadaten und Annotationen	365
8.3.4	ABAP-RESTful-Programmiermodell	366
8.3.5	SAP List Viewer mit integriertem Datenzugriff	370
8.4	Umbau des Custom Codes auf neue Objekte	371
8.5	Erweiterungskonzepte und Clean Core	372
8.5.1	Klassische Erweiterbarkeit	375
8.5.2	In-App-Erweiterbarkeit	376
8.5.3	Side-by-Side-Erweiterbarkeit	379
8.5.4	Clean Core und SAP S/4HANA Cloud	381
8.6	Zusammenfassung	382

9	Best Practices für Transformationsprojekte	385
9.1	Typische Herausforderungen und Fallstricke	385
9.1.1	Organisatorische Herausforderungen und Fallstricke	385
9.1.2	Technische Herausforderungen und Fallstricke	391
9.2	Spezielle Lösungsansätze und zusätzliche Tools	400
9.2.1	Umgang mit gelöschten SAP-ERP-Datentypen	400
9.2.2	Selektive Migration von Objekten	405
9.2.3	Programmmengenverarbeitung mit Transaktion SAMT	407
9.2.4	Prüfung von Bedingungen und Formeln in SD	409
9.2.5	Nutzung der Korrektur-Workbench	410
9.3	Partnertools und -services	416
9.3.1	SNP	417
9.3.2	All for One Group SE	418
9.3.3	Swisscom SAP Services	419
9.3.4	smartShift	419
9.4	Zusammenfassung	420
	Anhang	423
A	Wichtige Transaktionen	425
B	Checkliste zur Vorbereitung der Custom-Code-Migration	429
C	Das Autorenteam	431
	Index	433