

SAP PRESS

ABAP-Programmierung für die SAP-Finanzbuchhaltung – Kundeneigene Erweiterungen

User-Exits und BAdIs für SAP FI

Bearbeitet von
Michael Rohrbach

1. Auflage 2012. Buch. ca. 385 S. Hardcover
ISBN 978 3 8362 1863 4
Format (B x L): 16 x 24 cm

[Weitere Fachgebiete > EDV, Informatik > Datenbanken, Informationssicherheit,
Geschäftssoftware > SAP](#)

schnell und portofrei erhältlich bei


DIE FACHBUCHHANDLUNG

Die Online-Fachbuchhandlung beck-shop.de ist spezialisiert auf Fachbücher, insbesondere Recht, Steuern und Wirtschaft. Im Sortiment finden Sie alle Medien (Bücher, Zeitschriften, CDs, eBooks, etc.) aller Verlage. Ergänzt wird das Programm durch Services wie Neuerscheinungsdienst oder Zusammenstellungen von Büchern zu Sonderpreisen. Der Shop führt mehr als 8 Millionen Produkte.

Michael Rohrbach

ABAP®-Programmierung für die SAP®-Finanzbuchhaltung – Kundeneigene Erweiterungen



Galileo Press 

Bonn • Boston

Inhalt

Einleitung	11
------------------	----

1 Allgemeines zu kundeneigenen Erweiterungen 15

1.1	Verwendung von SAP-Erweiterungen (SMOD/CMOD)	15
1.1.1	SAP-Erweiterungen anzeigen	16
1.1.2	SAP-Erweiterungen implementieren	18
1.1.3	SAP-Erweiterungen aktivieren und deaktivieren	25
1.1.4	SAP-Erweiterungen testen	25
1.1.5	SAP-Erweiterungen finden	26
1.2	Verwendung von Business Transaction Events	27
1.2.1	Verwendung und Gruppierung	28
1.2.2	Business Transaction Events anzeigen	30
1.2.3	Business Transaction Events implementieren	32
1.2.4	Business Transaction Events konfigurieren	34
1.2.5	Business Transaction Events testen	35
1.2.6	Technische Details zu Business Transaction Events	36
1.2.7	Business Transaction Events finden	36
1.3	Verwendung von klassischen Business Add-Ins	38
1.3.1	Klassische Business Add-Ins anzeigen	39
1.3.2	Klassische Business Add-Ins implementieren	40
1.3.3	Klassische Business Add-Ins aktivieren und deaktivieren	43
1.3.4	Klassische Business Add-Ins testen	43
1.3.5	Klassische Business Add-Ins finden	44
1.4	Verwendung von neuen Business Add-Ins	45
1.4.1	Neue Business Add-Ins anzeigen	45
1.4.2	Neue Business Add-Ins implementieren	47
1.4.3	Neue Business Add-Ins aktivieren und deaktivieren	49
1.4.4	Neue Business Add-Ins finden	49
1.5	Verwendung von impliziten Erweiterungen	49
1.6	Zusammenfassung	50

TEIL I: Kundeneigene Erweiterungen von Stammdaten

2 Sachkontenstammdaten 53

2.1	Technische Details und Transaktionen	53
2.2	Ziel der Erweiterung der Sachkontenstammdaten	55

2.3	Erweiterung der Datenbanktabellen und Strukturen	56
2.3.1	Datenelement und Domäne für das Zusatzfeld	56
2.3.2	Wertetabelle für das Zusatzfeld	58
2.3.3	Langtexte zu Wertetabelle für das Zusatzfeld	60
2.3.4	Pflege-View zu Wertetabelle für das Zusatzfeld	62
2.3.5	Erweiterung der Tabelle SKB1	64
2.3.6	Erweiterung der Verarbeitungsstrukturen	65
2.4	Dialogerweiterung in der Sachkontenstammdatenpflege	68
2.4.1	Subscreen für das Zusatzfeld anlegen	69
2.4.2	Integration des Screens für das Zusatzfeld	71
2.4.3	Definition der Verarbeitungslogik für das Zusatzfeld	76
2.5	Zusätzliche Prüfungen für Sachkontenstammdaten	80
2.6	Schnittstellen für Sachkontenstammdaten	83
2.7	Zusammenfassung	84

3 Debitorenstammdaten 85

3.1	Technische Details und Transaktionen	85
3.2	Ziel der Erweiterung der Debitorenstammdaten	86
3.3	Erweiterung der Datenbanktabellen und Strukturen	87
3.4	Dialogerweiterung in der Debitorenstammdatenpflege	88
3.4.1	Subscreen für das Zusatzfeld anlegen	89
3.4.2	Bildgruppe zu Subscreen definieren	90
3.4.3	Bildgruppe aktivieren und Subscreen zuordnen	91
3.4.4	Definition der Verarbeitungslogik für das Zusatzfeld	97
3.5	Zusätzliche Prüfungen für Debitorenstammdaten	103
3.6	Schnittstellen für Debitorenstammdaten	104
3.6.1	Erweiterung der Batch-Input-Struktur	107
3.6.2	Erweiterung des Batch-Input-Programms	108
3.7	Zusammenfassung	111

4 Kreditorenstammdaten 113

4.1	Technische Details und Transaktionen	114
4.2	Ziel der Erweiterung der Kreditorenstammdaten	114
4.3	Erweiterung der Datenbanktabellen und Strukturen	115
4.4	Dialogerweiterung in der Kreditorenstammdatenpflege	117
4.4.1	Definition der Verarbeitungslogik für das Zusatzfeld	121
4.4.2	Fortschreibung von Änderungsbelegen für das Zusatzfeld	127
4.4.3	Anzeige von Änderungsbelegen für das Zusatzfeld	131

4.5	Zusätzliche Prüfungen für Kreditorenstammdaten	133
4.6	Schnittstellen für Kreditorenstammdaten	135
4.7	Zusammenfassung	136

TEIL II: Kundeneigene Erweiterungen von Geschäftsvorfällen

5	Buchhaltungsbelege	139
5.1	Technische Details der Buchhaltungsbelege	139
5.2	Transaktionen	146
5.3	Dialogerweiterungen von Buchungstransaktionen	147
5.3.1	Zusätzliche Funktionscodes in der Erfassung von Belegpositionen	147
5.3.2	Pflege zusätzlicher Belegkopfdaten in Enjoy- Transaktion für Sachkontenbelege	152
5.3.3	Pflege zusätzlicher Belegkopfdaten in Enjoy- Transaktion für Debitoren- und Kreditorenbelege	158
5.4	Kundeneigene Zusatzkontierungen in Buchhaltungsbelegen	161
5.5	Erweiterungen der Belegbuchung	166
5.6	Schnittstellen für Buchhaltungsbelege	167
5.6.1	Batch-Input bzw. Call Transaction	168
5.6.2	Direct Input	170
5.7	Validierungen und Substitutionen	171
5.7.1	Implementierung von User-Exits	172
5.7.2	Beispielsubstitution mit User-Exits	174
5.8	Zusammenfassung	177
6	Elektronischer Kontoauszug	179
6.1	Technischer Ablauf der Kontoauszugsverarbeitung	179
6.2	Programme und Transaktionen	183
6.3	Erweiterungen während des Einlesens der Kontoauszugsdaten	183
6.3.1	SWIFT-Standardformat MT940	183
6.3.2	Erweiterungen während des Einlesens des MT940-Formats	185
6.4	Erweiterungen während der Interpretation der Kontoauszugsdaten	188
6.5	Erweiterungen während der Buchung der Kontoauszugsdaten	192
6.6	Erweiterungen in der Nachbearbeitung von Kontoauszugsdaten	195

6.7	Interpretationsalgorithmen	200
6.7.1	Standard-Interpretationsalgorithmen	200
6.7.2	Kundeneigene Interpretationsalgorithmen	205
6.8	Suchmuster	210
6.9	Zusammenfassung	215
7	Mahnlauf	217
7.1	Technische Details und Ablauf des Mahnlaufs	217
7.2	Programme und Transaktionen	219
7.3	Erweiterungen des Mahnlaufs	220
7.3.1	Erweiterungen der Transaktion F150	221
7.3.2	Erweiterungen während der Mahnselektion	226
7.3.3	Erweiterungen während des Mahndrucks	234
7.4	Zusammenfassung	236
8	Zahlauf	237
8.1	Technische Details und Verlauf des Zahllaufs	237
8.2	Programme und Transaktionen	239
8.3	Erweiterungen des Zahllaufs	240
8.3.1	Erweiterungen der Transaktion F110	241
8.3.2	Erweiterungen des Zahlungsprogramms	243
8.3.3	Erweiterungen in der Regulierungsliste	250
8.4	Erweiterungen in der Payment Medium Workbench	253
8.4.1	Zeitpunktbausteine	254
8.4.2	Aufbau des Verwendungszwecks	257
8.4.3	Formatspezifische Parameter	261
8.4.4	Kundeneigene Granularität	263
8.5	Zusammenfassung	265
9	Reporting – Einzelpostenliste	267
9.1	Technische Details und Transaktionen	267
9.2	Erweiterungen der Einzelpostenliste in den Transaktionen FBL*N	269
9.2.1	Erweiterung der Kopfzeilen der Einzelpostenliste	269
9.2.2	Menüerweiterungen der Einzelpostenliste	273
9.2.3	Zusätzliche Felder in der Einzelpostenliste	280
9.3	Erweiterungen der Einzelpostenliste in der Transaktion FAGLLO3	284

9.3.1	Menüerweiterungen der Einzelpostenliste	285
9.3.2	Zusätzliche Felder in der Einzelpostenliste	285
9.4	Zusammenfassung	286

A Übersicht der Erweiterungen in der SAP-Finanzbuchhaltung 287

A.1	Sachkontenstammdaten	288
A.2	Stammdaten Debitoren-/Kreditorenbuchhaltung	289
A.3	Geschäftsvorfälle Debitoren-/Kreditorenbuchhaltung	298
A.4	Buchhaltungsbelege	301
A.5	Elektronischer Kontoauszug	319
A.6	Mahnlauf	324
A.7	Zahllauf	330
A.8	Reporting	334
A.9	Periodische Arbeiten	340
A.10	Meldewesen	349
A.11	Archivierung	354
A.12	IDoc-Verarbeitung	358
A.13	Länderspezifische Erweiterungen	365
A.14	Klassische Zahlungsträgerprogramme	370

B Der Autor 373

Index	375
-------------	-----

Einleitung

Die Finanzbuchhaltung als Teil des SAP ERP-Moduls Finanzwesen (FI) ist aufgrund der umfangreichen Customizing-Möglichkeiten auch ohne kundeneigene Erweiterungen sehr flexibel. Trotzdem gelangt man früher oder später an einen Punkt, an dem bloßes Customizing für die Umsetzung einer Anforderung nicht mehr ausreicht.

Bevor die verschiedenen SAP-Erweiterungstechniken eingeführt wurden, mussten in so einem Fall Standardprogramme modifiziert werden, was aufgrund des vollständig vorliegenden Quellcodes des SAP-Systems technisch auch kein Problem darstellte. Der Nachteil bei der Anpassung von Standardprogrammen bestand allerdings darin, dass solche Modifikationen zu Problemen bei Release-Wechseln führen konnten und SAP keine Verantwortung für das Verhalten von modifizierten Programmen übernimmt.

Um den Kunden trotzdem die Möglichkeit zu geben, die Verarbeitungslogik von Standardprogrammen anzupassen, ohne den Support-Anspruch zu verlieren, führte SAP nach und nach verschiedene Erweiterungstechniken ein. Seitdem ist es möglich, das SAP-System bei Bedarf an definierten Punkten modifikationsfrei zu erweitern und Daten über festgelegte Schnittstellen zu ändern oder sogar eigene Funktionen in den Programmablauf zu integrieren.

Zielsetzung

Dieses Buch vermittelt Ihnen zunächst die nötigen technischen Grundlagen für die Implementierung von kundeneigenen Erweiterungen. Anschließend sehen Sie verschiedene Anwendungsbeispiele für Erweiterungen von ausgewählten Stammdaten und Geschäftsvorfällen in der Finanzbuchhaltung. In detaillierten Schritt-für-Schritt-Anleitungen mit ABAP-Beispiel-Codings wird das Vorgehen zur Realisierung diverser Anforderungen gezeigt. Gegebenenfalls werden hier mehrere Alternativen vorgestellt. In Ergänzung zu diesen inhaltlich beschriebenen Erweiterungen finden Sie im Anhang eine vollständige Übersicht über alle Erweiterungsmöglichkeiten in der SAP-Finanzbuchhaltung.

Das Ziel dieses Buches ist dabei, Ihnen anhand der hier ausgewählten Beispiele aus der Hauptbuchhaltung sowie der Debitoren- und Kreditorenbuchhaltung eine systematische Herangehensweise an die Implementierung von Erweiterungen zu vermitteln. Gleichzeitig soll Ihnen das Buch als Nachschlagewerk für typische Erweiterungen in der Finanzbuchhaltung dienen.

Aufbau und Inhalt

Das Buch beginnt in **Kapitel 1**, »Allgemeines zu kundeneigenen Erweiterungen«, mit einer ausführlichen Einführung in die verschiedenen Arten von kundeneigenen Erweiterungen in der SAP-Finanzbuchhaltung. Anhand von einfachen Beispielen lernen Sie, wie Sie User-Exits, Business Transaction Events (BTE) und Business Add-Ins (BAI) verwenden und aktivieren. Außerdem sehen Sie, wie Sie selbst Erweiterungspunkte im System finden können, sei es über das Infosystem oder auf technischem Weg über ein Debugging der betreffenden Programme. Das Kapitel wird ergänzt um die Beschreibung von impliziten Erweiterungen.

Die folgenden Ausführungen in **Teil 1**, »Kundeneigene Erweiterungen von Stammdaten«, behandeln kundeneigene Erweiterungen von Stammdaten in der Finanzbuchhaltung. In **Kapitel 2**, »Sachkontenstammdaten«, **Kapitel 3**, »Debitorenstammdaten«, und **Kapitel 4**, »Kreditorenstammdaten«, erfahren Sie zunächst, wie diese Stammdaten im SAP-System abgebildet und welche Transaktionen für die Pflege verwendet werden. Anschließend sehen Sie, wie die Stammdaten um kundeneigene Daten erweitert und die zugehörigen Pflegedialoge modifikationsfrei angepasst werden können. Außerdem wird beschrieben, wie Schnittstellen für Stammdaten in der Finanzbuchhaltung realisiert werden und welche Erweiterungsmöglichkeiten existieren.

Die in **Teil 2**, »Kundeneigene Erweiterungen von Geschäftsvorfällen«, folgenden Kapitel beinhalten kundeneigene Erweiterungen von wichtigen Geschäftsprozessen in der Finanzbuchhaltung. Zunächst werden in **Kapitel 5**, »Buchhaltungsbelege«, mögliche Erweiterungen von Buchhaltungsbelegen und in **Kapitel 6**, »Elektronischer Kontoauszug«, Erweiterungen zum elektronischen Kontoauszug beschrieben. Weitere Geschäftsprozess-Erweiterungsmöglichkeiten lernen Sie in **Kapitel 7**, »Mahnlauf«, und in **Kapitel 8**, »Zahllauf«, kennen. Exemplarisch für das Reporting werden in **Kapitel 9**, »Reporting – Einzelpostenliste«, mögliche Erweiterungen der Einzelpostenliste dargestellt. Zu jedem der Geschäftsvorfälle werden zunächst die technischen Grundlagen erläutert und darauf aufbauend Erweiterungen der Verarbeitungslogik bzw. der Dialoge anhand von Praxisbeispielen dargestellt. Die Coding-Beispiele sind dabei

bewusst einfach gehalten, vermitteln Ihnen aber das nötige Wissen, um anschließend Ihre ganz individuellen Anforderungen selbst umsetzen zu können. Die Schwierigkeit einer Erweiterung liegt nämlich häufig nicht in der Implementierung selbst, sondern darin, das unterliegende Datenmodell bzw. die unterliegenden Prozesse zu verstehen.

Anhang A, »Übersicht der Erweiterungen in der SAP-Finanzbuchhaltung«, enthält schließlich eine Übersicht über alle User-Exits, Business Transaction Events und Business Add-Ins, die in der SAP-Finanzbuchhaltung am häufigsten zum Einsatz kommen. Zu jedem Erweiterungspunkt beinhaltet die Übersicht eine kurze Beschreibung der Funktion sowie die Komponenten und Aufrufstellen der entsprechenden Erweiterungsimplementierung, sodass Sie für Ihre eigenen Erweiterungsprojekte schnell die geeigneten Exits finden können.

Um Ihnen die Lektüre zu erleichtern, werden in diesem Buch die folgenden Symbole verwendet:

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

Mit diesem Symbol werden Tipps markiert, die Ihnen die Arbeit erleichtern können. Auch Empfehlungen, die Ihnen z. B. dabei helfen, weiterführende Informationen zu dem besprochenen Thema zu finden, werden mit diesem Symbol hervorgehoben. **[+]**

Wenn das besprochene Thema anhand von praktischen Beispielen erläutert und vertieft wird, werden Sie mit diesem Symbol darauf aufmerksam gemacht. **[zB]**

Zielgruppe

Dieses Buch richtet sich in erster Linie an ABAP-Entwickler, die einen Einstieg in die Erweiterungsmöglichkeiten der SAP-Finanzbuchhaltung suchen, ist aber auch für FI-Berater mit grundlegenden Programmierkenntnissen geeignet. Darüber hinaus können sich technische Projektleiter einen Überblick über die Erweiterungen verschaffen, ohne sich zu sehr in Implementierungsdetails vertiefen zu müssen.

Das Buch richtet sich daher an Leser mit unterschiedlichen fachlichen und technischen Kenntnissen und soll trotzdem jeder dieser Gruppen das nötige Wissen über Erweiterungsmöglichkeiten in der Finanzbuchhaltung vermit-

teln. Daher wird versucht, sowohl fachliche als auch technische Grundlagen, die für das Verständnis einer Erweiterung nötig sind, ausreichend detailliert zu erläutern. Das Hauptaugenmerk dieses Buch liegt aber – wie der Titel schon andeutet – auf der technischen Umsetzung der Erweiterungen. Die technische Umsetzung wird daher ausführlich beschrieben, während fachliche Zusammenhänge gegebenenfalls vereinfacht dargestellt werden.

Voraussetzungen

Für das Verständnis der Erweiterungsimplementierungen sind Grundkenntnisse in der ABAP-Programmierung erforderlich. Einige Erweiterungsmöglichkeiten sind außerdem objektorientiert umgesetzt, sodass entsprechende ABAP Objects-Kenntnisse von Vorteil sind. Diese sind allerdings keinesfalls ein Muss, da diese Erweiterungen durch die Benutzerführung im SAP-System auch implementiert werden können, ohne zu wissen, dass es sich im Kern um eine objektorientierte Umsetzung handelt.

Für alle Leser mit fachlichem Hintergrund, die sich nur einen Überblick über die Erweiterungsmöglichkeiten verschaffen möchten, sind die Kapitel außerdem so strukturiert, dass die Implementierungsdetails bei Bedarf auch übersprungen werden können.

Danksagung

Ich danke meinem Arbeitgeber, der ConVista Consulting AG, und hier speziell Oliver Kewes, für die entgegenkommende Arbeitszeitregelung, die dieses Buch überhaupt erst ermöglicht hat. Außerdem danke ich meinem Kollegen Samuel Gonzalez, der viele Stunden seiner Freizeit für Anregungen, Verbesserungsvorschläge und Korrekturen geopfert hat. Darüber hinaus möchte ich mich bei Janina Schweitzer und Stefan Proksch von Galileo Press für das Lektorat und die Unterstützung während der Manuskripterstellung bedanken.

Der größte Dank gilt allerdings meiner Frau, die mir besonders in den letzten Wochen privat jede erdenkliche Arbeit abgenommen hat und mir so die nötige Zeit verschafft hat, dieses Buch pünktlich fertigzustellen.

Michael Rohrbach

5 Buchhaltungsbelege

Buchhaltungsbelege sind das zentrale Objekt für Bewegungsdaten in der Finanzbuchhaltung, denn jede Bewegung auf Sach-, Debitoren- und Kreditorenkonten erfordert die Buchung eines Belegs. Grundlage dieses Kapitels bilden die Darstellung der verwendeten Datenbanktabellen für die Ablage von Buchhaltungsbelegen sowie die verschiedenen Transaktionen für deren Erfassung, Änderung und Anzeige. Im Anschluss an die Darstellung der Grundlagen sehen Sie, welche Möglichkeiten Sie haben, Buchungsdialo-ge anzupassen oder sogar in die Belegerstellung selbst einzugreifen. Außerdem finden Sie Informationen zur Erweiterung von Schnittstellen, sowohl aus anderen Modulen als auch aus externen Quellen. Das Kapitel schließt mit Erläuterungen zur Verwendung von User-Exits in Validierungen und Substitutionen.

5.1 Technische Details der Buchhaltungsbelege

Grundsätzlich ist die Struktur eines Buchhaltungsbeleges sehr einfach und benötigt eigentlich nur zwei Datenbanktabellen: eine für den Belegkopf und eine, die die Belegpositionen enthält. Wie Sie in diesem Abschnitt sehen werden, existieren jedoch wesentlich mehr Tabellen, die teilweise redundante Informationen enthalten. Diese zusätzlichen Tabellen ermöglichen aber z. B. die performante Erstellung von Steuermeldungen oder Einzelposten-, Offene-Posten- (OP) und Saldenlisten, die ohne sie nicht möglich wären.

Belegkopf (Tabelle BKPF)

Belegköpfe werden in der Tabelle BKPF abgelegt und enthalten positionsübergreifende Informationen wie das Beleg- und Buchungsdatum, die Belegart oder die Währung eines Belegs. Ein Belegkopf wird durch die Kombination von Buchungskreis, Belegnummer und Geschäftsjahr eindeutig identifiziert.

Belegpositionen (Tabelle BSEG)

Die Tabelle BSEG enthält alle Positionen eines Belegs und ist über die Felder Buchungskreis, Belegnummer und Geschäftsjahr mit der Tabelle BKPF verknüpft. Die Tabelle BSEG enthält im Schlüssel zusätzlich noch einen dreistelligen Zähler für die Buchungszeile.



Belegpositionen bei Schnittstellenimplementierung

Ein Beleg kann demnach maximal 999 Positionen enthalten, was gerade bei der Implementierung von Schnittstellen berücksichtigt werden muss. Dabei ist außerdem zu beachten, dass vom System z. B. bei Steuerbuchungen oder buchungskreisübergreifenden Buchungen automatisch zusätzliche Zeilen erzeugt werden, die ebenfalls noch in den Beleg passen müssen.

Die Tabelle BSEG enthält über 300 Datenfelder, von denen die wichtigsten hier kurz dargestellt und erläutert werden:

► Buchungsschlüssel (BSCHL)

Der Buchungsschlüssel beschreibt die Art der Bewegung in einer Belegposition. Aus dem Buchungsschlüssel ergeben sich unter anderem die bebuchte Kontoart und die Zuordnung zur Soll- oder Haben-Seite. Außerdem ist es möglich, für den Buchungsschlüssel einen Feldstatus zu hinterlegen, das heißt zu definieren, welche Eingaben bei der Verwendung eines Buchungsschlüssels vorgenommen werden können oder müssen. Dies kann z. B. bei debitorischen und kreditorischen Positionen sinnvoll sein, da hier eine differenzierte Ausgestaltung des Feldstatus nicht immer über das zugehörige Hauptbuchkonto erfolgen kann.

► Kontoart (KOART)

Die Kontoart legt fest, ob ein Sachkonto (S), ein Debitorenkonto (D) oder ein Kreditorenkonto (K) in der Buchungszeile bebucht wird. Daneben gibt es noch Kontoarten für Buchungen in der Anlagenbuchhaltung und Materialwirtschaft, auf die hier nicht näher eingegangen wird.

► Soll-/Haben-Kennzeichen (SHKZG)

Das Soll-/Haben-Kennzeichen legt fest, ob es sich um eine Soll- oder Haben-Position handelt, was letztlich für die korrekte Fortschreibung der Summentabellen (siehe Abschnitt »Summentabellen des Hauptbuchs (GLT0 und FAGLFLEXT)«) von Bedeutung ist.

► Betrag in Hauswährung (DMBTR)

Der Hauswährungsbetrag enthält den Betrag in der Währung, die als Hauswährung für den Buchungskreis definiert wurde.

- ▶ **Betrag in Transaktionswährung (WRBTR)**
Der Transaktionswährungsbetrag enthält den Betrag in der Währung, die im Belegkopf festgelegt wurde.
- ▶ **Konto (HKONT, KUNNR, LIFNR)**
Abhängig von der Kontoart, finden Sie das gebuchte Konto in unterschiedlichen Feldern: bei Sachkontenpositionen im Feld `HKONT`, bei Debitorenpositionen im Feld `KUNNR` und bei Kreditorenpositionen im Feld `LIFNR`. Bei Debitoren- oder Kreditorenpositionen ist außerdem zusätzlich das Feld `HKONT` gefüllt und enthält das Abstimmkonto des Debitors- bzw. Kreditors, da die Buchung auch auf diesem Konto fortgeschrieben wird.
- ▶ **Steuerkennzeichen (MWSKZ)**
Das Steuerkennzeichen legt fest, ob eine Belegposition steuerrelevant ist, und bestimmt die Art (z. B. Unterscheidung zwischen Vorsteuer und Ausgangsteuer) und Höhe der Steuer.
- ▶ **Positionstext (SGTXT)**
Im Positionstext kann zu jeder Belegposition ein beschreibender Text erfasst werden.
- ▶ **Zuordnungsnummer (ZUONR)**
Die Zuordnungsnummer ist ein weiteres Freitextfeld, das als Zusatzinformation am Beleg häufig für die Sortierung in Einzelpostenlisten oder zur Gruppierung von Posten beim OP-Ausgleich verwendet wird.
- ▶ **CO-Kontierungen (KOSTL, AUFNR etc.)**
Bei der Buchung auf Kostenarten enthält die Belegposition das oder die zugehörigen CO-Kontierungsobjekte.
- ▶ **Partnergesellschaft (VBUND)**
Die Partnergesellschaft ist eine konzernweit eindeutige Gesellschaftsnummer und wird in der Regel dazu verwendet, Forderungen, Verbindlichkeiten, Aufwände und Erträge einer Gesellschaft einer anderen Gesellschaft des Konzerns zuzuordnen. Dadurch wird zum Beispiel die Eliminierung konzerninterner Forderungen und Verbindlichkeiten für die Erstellung einer Konzernbilanz ermöglicht.

Technisch ist die Tabelle `BSEG` eine Cluster-Tabelle und zusammen mit anderen Tabellen einem Tabellen-Cluster (`RFBLG`) zugeordnet. Das heißt, dass die Tabelle nicht wie transparente Tabellen eins zu eins auch in der Datenbank existiert. Stattdessen enthält der Tabellen-Cluster in der Datenbank nur alle gemeinsamen Schlüsselfelder der enthaltenen Cluster-Tabellen und alle anderen Felder werden komprimiert in einer einzigen Spalte abgelegt. Daraus ergibt sich zum einen die Einschränkung, dass die Tabelle `BSEG` nicht in

SQL-Joins verwendet werden darf, und außerdem ist ein performanter Zugriff nur über den Tabellenschlüssel möglich.

CPD-Daten (Tabelle BSEC)

Bei der Erfassung von Belegpositionen zu CPD-Debitoren oder -Kreditoren (CPD steht für Conto pro Diverse) werden zusätzliche Buchungsinformationen wie Adressen oder Bankverbindungen auf Belegpositionsebene in der Tabelle BSEC gespeichert.

Buchungskreisübergreifende Buchungen (Tabelle BVOR)

Bei buchungskreisübergreifenden Buchungen wird pro beteiligten Buchungskreis ein normaler Beleg mit den entsprechenden Einträgen in den Tabellen BKPF, BSEG etc. erstellt. Gleichzeitig wird eine Nummer vergeben, die eine übergreifende Klammer über alle Einzelbelege in den Buchungskreisen bildet und in der Tabelle BVOR gespeichert wird.

Steuerdaten zur Belegposition (Tabelle BSET)

Die Tabelle BSET ist ebenfalls eine Cluster-Tabelle im Tabellen-Cluster `RFBLG` und enthält Steuerpositionen. Steuerpositionen können sowohl manuell erfasst als auch automatisch vom System erzeugt werden. Bei der Erzeugung von Steuerzeilen werden alle steuerrelevanten Buchungszeilen nach Steuerkennzeichen gruppiert, dann wird für jedes Steuerkennzeichen ein resultierender Steuerbetrag berechnet und das Ergebnis als einzelne Steuerzeile hinzugefügt. Im Allgemeinen gibt es also keine Eins-zu-eins-Beziehung zwischen einer steuerrelevanten Zeile und einer Steuerzeile.

Sekundärindextabellen (Tabellen BS*)

Sekundärindextabellen enthalten redundante Informationen aus den Tabellen BKPF und BSEG und ermöglichen die performante Erstellung von OP- und Einzelpostenlisten. Wie bereits erwähnt, kann auf die Tabelle BSEG nur über den Schlüssel, das heißt über Belege, performant zugegriffen werden. Bei der Erstellung von OP- und Einzelpostenlisten wird aber ein Zugriff über Kontonummern bzw. Ausgleichsinformationen von Positionen benötigt.

Daher gibt es verschiedene Tabellen, die neben dem Schlüssel der Tabelle BSEG die Kontonummer, das Ausgleichsdatum, den Ausgleichsbeleg und die Zuordnung enthalten und verschiedene Datenfelder aus BKPF und BSEG für

Auswertungen zusammenfassen. Insgesamt gibt es sechs solcher Sekundärindextabellen, die alle mit dem Präfix **BS** beginnen. Es gibt unterschiedliche Tabellen für offene und ausgeglichene Posten sowie für Sachkonten-, Debitoren- und Kreditorenposten. Der Tabellenname ergibt sich in Ergänzung zum Präfix **BS** wie folgt: Der dritte Buchstabe des Tabellennamens unterscheidet zwischen offenen (I) und ausgeglichenen (A) Posten, und der vierte Buchstabe unterscheidet zwischen Sachkonten (S), Debitoren (D) und Kreditoren (K). Daraus ergeben sich die folgenden Sekundärindextabellen:

- ▶ BSIS und BSAS für Sachkontenpositionen
- ▶ BSID und BSAD für Debitorenpositionen
- ▶ BSIK und BSAK für Kreditorenpositionen

Besonderheit der Tabelle BSIS

Der Inhalt der Tabelle BSIS unterscheidet sich dabei etwas vom Inhalt der Tabellen BSID und BSIK – von dem Unterschied in der betreffenden Kontoart einmal abgesehen. Während Debitoren und Kreditoren immer OP-geführt sind, kann ein Sachkonto OP- oder einzelpostengeführt sein, oder es werden gar keine Einzelposten für das Sachkonto fortgeschrieben. Die Tabelle BSIS wird allerdings nicht nur bei OP-Führung des Kontos fortgeschrieben, sondern auch bei Einzelpostenführung, um die Erstellung von Einzelpostenlisten zu ermöglichen. Die Tabelle BSIS enthält also nicht ausschließlich offene Posten, sondern für einzelpostengeführte Konten alle Positionen. Positionen von OP-geführten Konten können dabei über das Kennzeichen **XOPVW** identifiziert werden.

[+]

Beim Ausgleich eines Sachkonten-, Debitoren- oder Kreditoren-OPs wird die zugehörige Tabellenzeile aus der Tabelle BSI* entfernt und in die Tabelle BSA* eingefügt. Dabei werden das Ausgleichsdatum und die Ausgleichsbelegnummer gefüllt. Ein Posten eines OP-geführten Sachkontos, eines Debitors oder eines Kreditors kann demnach immer nur entweder in der Tabelle BSI* oder der Tabelle BSA* enthalten sein, wobei das Ausgleichsdatum und die Ausgleichsbelegnummer in den Tabellen BSI* immer leer und in den Tabellen BSA* immer gefüllt sind.

Bei der Auswertung von offenen Posten über die Tabellen BSI* und BSA* ist zu beachten, dass die Tabelle BSI* immer den *aktuellen* Stand von offenen Posten enthält. Mit den Informationen aus beiden Tabellen lassen sich jedoch die offenen Posten zu jedem beliebigen Stichtag in der Vergangenheit bestimmen. Dazu müssen folgende Positionen aus BSI* und BSA* zusammengefasst werden:

- ▶ alle Posten aus BSI* mit Buchungsdatum kleiner gleich dem Stichtag
- ▶ alle Posten aus BSA* mit Buchungsdatum kleiner gleich dem Stichtag und Ausgleichsdatum größer gleich dem Stichtag

Besonderheiten bei der Verwendung des neuen Hauptbuches

Bei Verwendung des neuen Hauptbuches (New General Ledger, kurz New GL) werden noch in weitere Tabellen Belegpositionen bzw. Einzelposten fortgeschrieben:

- ▶ Die Tabelle BSEG_ADD enthält alle Belegzeilen von Belegen, die in nicht führenden Ledgern gebucht werden. Da bei der Buchung in nicht führende Ledger einige Funktionen nicht unterstützt werden, enthält die Tabelle BSEG_ADD wesentlich weniger Datenfelder als die Tabelle BSEG. Als Beispiel seien hier alle Felder zu Buchungen auf Debitoren und Kreditoren genannt.
- ▶ Des Weiteren werden alle Einzelposten ledgergruppenspezifisch in der Tabelle FAGLFLEXA fortgeschrieben.
- ▶ Seitdem die Möglichkeit besteht, Sachkonten auch ledgergruppenspezifisch auszugleichen, gibt es außerdem die Tabellen FAGLBSIS und FAGLBSAS, die in Aufbau und Funktion grundsätzlich den Tabellen BSIS und BSAS entsprechen und die offenen und ausgeglichenen Posten von ledgergruppenspezifisch OPgeführten Konten enthalten. Im Gegensatz zur Tabelle BSIS enthält die Tabelle FAGLBSIS allerdings keine Posten von einzelpostengeführten Konten.

Summentabellen des Hauptbuches (GLT0 und FAGLFLEXT)

In der Finanzbuchhaltung werden häufig Auswertungen zu Sachkontensalden sowie natürlich zur Bilanz und Gewinn- und Verlustrechnung (GuV) benötigt. Technisch wäre es problemlos möglich, Kontensalden über die Einzelpostentabellen (BSI*) oder sogar über die Belegtabellen (BKPF und BSEG) zu berechnen. Zum Beispiel für die Erstellung einer Eröffnungsbilanz müssten dann allerdings alle jemals in der Vergangenheit gebuchten Posten bzw. Belege ausgewertet werden. Daher werden bereits bei der Buchung im Hintergrund Tabellen mit gebuchten Summen pro Periode und Saldovortrag fortgeschrieben. Für eine Eröffnungsbilanz muss so nur noch der Saldovortrag selektiert werden bzw. für eine Schlussbilanz der Saldovortrag und die Periodenwerte des laufenden Jahres.

Die verwendete Summentabelle ist abhängig davon, ob New GL im System aktiviert ist oder nicht. Die grundlegende Tabellenstruktur ist identisch und wird daher im Folgenden anhand der Summentabelle des klassischen Hauptbuches beschrieben. Anschließend wird auf die Unterschiede eingegangen, die bei der Verwendung der neuen Hauptbuchbuchhaltung zu beachten sind.

Ist New GL nicht aktiviert, wird die Tabelle GLT0 als Summentabelle für das Hauptbuch verwendet. Darin werden die Verkehrszahlen in Abhängigkeit von Buchungskreis, Geschäftsjahr, Konto, Geschäftsbereich, Währung und Soll-/Haben-Kennzeichen fortgeschrieben. Maximal können in der Tabelle GLT0 Periodenwerte für 16 Perioden gespeichert werden: Hauswährungswerte finden sich in den Spalten HSL01 bis HSL16, Transaktionswährungswerte in den Spalten TSL01 bis TSL16.

Außerdem gibt es noch die Periodenwerte KSL01 bis KSL16 für eine sogenannte *Konzernwährung* bzw. *parallel geführte Hauswährung*. Diese Periodenwerte werden zum Beispiel verwendet, wenn innerhalb eines Konzerns Buchungskreise mit unterschiedlichen Hauswährungen existieren, aber eine Konzernbilanz in einer Währung erstellt werden soll. Neben den Periodenwerten enthält die Tabelle GLT0 die Saldovorträge des Geschäftsjahres in Hauswährung (Feld HSLVT), Transaktionswährung (Feld TSLVT) und Konzernwährung (Feld KSLVT). Der Wert des Saldovortrages entspricht genau der Summe aus Saldovortrag und allen Periodenwerten des Vorjahres.

Verwendung der Periodenwerte

[+]

Der Inhalt des Feldes RPMAX gibt an, wie viele der 16 Periodenwerte in der Tabelle GLT0 tatsächlich verwendet werden. In den meisten Fällen wird im SAP-System eine Geschäftsjahresvariante mit 16 Perioden verwendet, wobei die ersten zwölf Perioden den Monaten des Geschäftsjahres entsprechen und die letzten vier sogenannten *Sonderperioden* für Buchungen im Rahmen des Jahresabschlusses verwendet werden.

Für diese Geschäftsjahresvariante werden daher alle 16 Periodenwerte benötigt. Es können aber auch Geschäftsjahresvarianten verwendet werden, die weniger als vier Sonderperioden enthalten und damit nicht alle 16 Periodenwerte erfordern. Aus diesem Grund sollte in Summentabellen das Feld RPMAX bei Berechnungen immer berücksichtigt werden. Außerdem sollten Sie bei der Ableitung der Buchungsperiode aus dem Buchungsdatum stets den Funktionsbaustein DATE_TO_PERIOD_CONVERT verwenden, falls die Geschäftsvariante ein verschobenes Geschäftsjahr besitzt und daher der Monat des Jahres nicht mit der Periode übereinstimmt.

Bei aktiviertem New GL wird standardmäßig die Summentabelle FAGL-FLEXT verwendet, die im Hinblick auf die Periodenwerte der Tabelle GLT0 entspricht, allerdings die Fortschreibung von Verkehrszahlen für mehrere Dimensionen erlaubt. So ist es zum Beispiel möglich, Periodenwerte in Abhängigkeit von Kostenstellen, Partnergesellschaften oder sogar kundeneigenen Kontierungen fortzuschreiben.

Summentabellen der Nebenbücher (KNC1 und LFC1)

Auch in den Nebenbüchern werden Summentabellen fortgeschrieben, um die performante Auswertung von Debitoren- (KNC1) und Kreditorensalden (LFC1) zu ermöglichen. Im Gegensatz zum Soll-/Haben-Kennzeichen in den Summentabellen im Hauptbuch ist das Soll-/Haben-Kennzeichen in den Summentabellen des Nebenbuches kein Schlüsselfeld. Stattdessen sind Soll- und Haben-Werte sowie der resultierende Umsatz gemeinsam in derselben Tabellenzeile zu finden.

5.2 Transaktionen

Grundsätzlich kann bei der Erfassung von Buchhaltungsbelegen zwischen den klassischen Transaktionen und den neueren, sogenannten *Einbildtransaktionen* oder *Enjoy-Transaktionen* unterschieden werden. Alle klassischen Buchungstransaktionen basieren auf der Transaktion FB01, bei der Belegkopfinformationen und Belegpositionen nacheinander auf verschiedenen Dynpros eingegeben werden können. Abhängig vom Anwendungsgebiet, gibt es weitere Transaktionen, die die Transaktion FB01 starten und dabei verschiedene Felder schon mit Vorschlagswerten vorbelegen.

[zB]

Transaktion F-22

So gibt es z. B. im Bereich der Debitorenbuchhaltung die Transaktion F-22 (Debitorenrechnung erfassen), die die Transaktion FB01 mit der voreingestellten Belegart DR und dem Buchungsschlüssel 01 für die erste Belegposition startet.

In den Enjoy-Transaktionen können seit SAP-Release 4.6 alle Daten eines Geschäftsvorfalles gemeinsam auf einem Dynpro erfasst werden. Die gebräuchlichsten sind die Transaktionen für Sachkontenbuchungen (FB50), für die Buchung von Debitorenrechnungen (FB60) und für die Buchung von Kreditorenrechnungen (FB70). Allen Transaktionen ist gemeinsam, dass im oberen Bereich des Hauptbildes Belegkopfinformationen und in einer Tabelle im unteren Bereich des Hauptbildes Belegpositionen für Sachkonten eingegeben werden können. Darüber hinaus kann bei der Erfassung von Debitoren- und Kreditorenrechnungen genau eine Debitoren- bzw. Kreditorenposition zusammen mit den Belegkopfinformationen eingegeben werden. Als Beispiel ist die Enjoy-Transaktion zur Sachkontenbuchung in Abbildung 5.1 zu sehen. Enjoy-Transaktionen bieten im Vergleich zu klassischen Buchungstransaktionen mehr Erweiterungsmöglichkeiten.

Abbildung 5.1 Enjoy-Transaktion für Sachkontenbuchung

Sämtliche Buchungstransaktionen, gleichgültig, ob klassische oder Enjoy-Transaktion, sind im Programm SAPMF05A implementiert. Außerdem existiert das Programm SAPMF05L für die Beleganzeige und -änderung, die über die Transaktionen FB02 und FB03 aufgerufen werden.

5.3 Dialogerweiterungen von Buchungstransaktionen

Die Buchungsdialoge können sowohl bei klassischen als auch bei Enjoy-Transaktionen erweitert werden, wobei für Enjoy-Transaktionen mehr Erweiterungsmöglichkeiten zur Verfügung stehen.

5.3.1 Zusätzliche Funktionscodes in der Erfassung von Belegpositionen

Wenn Sie sich beispielhaft den GUI-Status ZBV im Programm SAPMF05A anschauen, sehen Sie, dass darin verschiedene Funktionscodes (OPF*) mit dynamischen Funktionstexten definiert sind. Dieser GUI-Status wird zum Beispiel im Dynpro 0300 bei der Erfassung einer Sachkontenposition verwendet. Anhand der Namen der Funktionscodes können Sie darauf schließen, dass es sich dabei um Open-FI-Erweiterungen handelt, die mit Business Transaction Events (BTE) aktiviert werden können.

Suchen Sie im BTE-Infosystem nach passenden Events, finden Sie unter anderem zwei P&S-Schnittstellen für die Aktivierung eines zusätzlichen Funktionscodes bei der Bearbeitung einer Belegposition:

► **P&S-Schnittstelle 00001080**

Das BTE 00001080 verwenden Sie, um einem der zusätzlichen Funktionscodes einen Funktionstext zuzuweisen und damit einzublenden.

► **P&S-Schnittstelle 00001070**

Das BTE 00001070 wird ausgeführt, sobald der durch das BTE 00001080 aktivierte Funktionscode ausgeführt wird. Dabei werden die Informationen zur aktuell bearbeiteten Buchungszeile übergeben.

In klassischen Transaktionen wird der Funktionscode nur in der Positionserfassung angezeigt, bei der Rechnungs- bzw. Gutschrifterfassung in Enjoy-Transaktionen sowohl in der Übersicht als auch in den Positionen. In der Übersicht werden bei der Ausführung des Funktionscodes die Informationen zur letzten Zeile in der Positionstabelle übergeben. Ist die Positionstabelle leer, wird die Funktion für den erfassten Debitor oder Kreditor ausgeführt.

In diesem Abschnitt sollen die genannten BTEs verwendet werden, um aus der Buchungstransaktion in die OP-Liste des aktuell erfassten Kontos abzuspriegen. Wie bei BTEs üblich gibt es Musterfunktionsbausteine, die kopiert und implementiert werden müssen. Anschließend werden die implementierten Funktionsbausteine den BTEs zugeordnet. Gehen Sie dazu wie folgt vor:

1. Kopieren Sie den Funktionsbaustein `SAMPLE_INTERFACE_00001080`, und erstellen Sie einen neuen Funktionsbaustein `Z_SAMPLE_INTERFACE_00001080` mit dem Quellcode aus Listing 5.1.

Im Quellcode wird unabhängig von Sprache und Bearbeitungsmodus der Funktionstext »OPs anzeigen« zurückgegeben, der als zusätzliche Drucktaste bzw. Menüeintrag erscheinen soll.

```
FUNCTION z_sample_interface_00001080.
*"-----
*"**"Lokale Schnittstelle:
*"  IMPORTING
*"    REFERENCE(I_SPRAS) LIKE  SY-LANGU
*"    REFERENCE(I_AKTYP) TYPE  AKTYP
*"    REFERENCE(I_DYNCL) TYPE  DYNCL
```

```

*" EXPORTING
*"     VALUE(E_FTEXT) LIKE  FTEXTS-FTEXT
*"-----
    e_ftext = 'OPs anzeigen'.
ENDFUNCTION.

```

Listing 5.1 Implementierung des BTEs 00001080

2. Kopieren Sie den Funktionsbaustein `SAMPLE_INTERFACE_00001070`, und erstellen Sie einen neuen Funktionsbaustein `Z_SAMPLE_INTERFACE_00001070` mit dem Quellcode aus Listing 5.2.

```

FUNCTION Z_SAMPLE_INTERFACE_00001070.
*"-----
**"*"Lokale Schnittstelle:
*" IMPORTING
*"     REFERENCE(I_BKPF) TYPE  BKPF
*"     REFERENCE(I_BSEG) TYPE  BSEG
*"     REFERENCE(I_AKTYP) TYPE  AKTYP
*"     REFERENCE(I_DYNCL) TYPE  DYNCL
*" EXPORTING
*"     REFERENCE(E_XCHNG) LIKE  OFIWA-XCHNG
*"-----
    SET PARAMETER ID 'BUK' FIELD i_bkpf-bukrs.
    IF i_bseg-koart EQ 'S'.
        SET PARAMETER ID 'SAK' FIELD i_bseg-hkont.
        CALL TRANSACTION 'FAGLL03' AND SKIP FIRST SCREEN.
    ELSEIF i_bseg-koart EQ 'K'.
        SET PARAMETER ID 'LIF' FIELD i_bseg-lifnr.
        CALL TRANSACTION 'FBL1N' AND SKIP FIRST SCREEN.
    ELSEIF i_bseg-koart EQ 'D'.
        SET PARAMETER ID 'KUN' FIELD i_bseg-kunnr.
        CALL TRANSACTION 'FBL5N' AND SKIP FIRST SCREEN.
    ELSE.
        * Gegebenenfalls Fehlermeldung:
        * OP-Anzeige für Kontoart & nicht möglich
    ENDIF.
ENDFUNCTION.

```

Listing 5.2 Implementierung des BTEs 00001070

Dem Funktionsbaustein des BTEs 00001070 werden Belegkopf- und Belegzeileninformationen der aktuell bearbeiteten Zeile übergeben. Abhängig von der bearbeiteten Kontoart in der Buchungszeile, wird die entsprechende Transaktion zur Anzeige einer OP-Liste aufgerufen und das Konto

über SPA-/GPA-Parameter übergeben. Um den Code möglichst übersichtlich zu halten, wurde hier auf jegliche Fehlerbehandlung verzichtet (die bei nicht OP-geführten Sachkonten notwendig würde).

[+]

SPA-/GPA-Parameter

Mithilfe von SPA-/GPA-Parametern können Werte transaktionsübergreifend zwischen Programmen ausgetauscht werden. Der Name leitet sich aus den ABAP-Anweisungen SET PARAMETER und GET PARAMETER ab, die verwendet werden, um Parameterwerte zu setzen oder abzufragen. In vielen Standardtransaktionen sind Parameter des Selektionsbildschirms mit SPA-/GPA-Parametern verknüpft, sodass diese in aufrufenden Programmen verwendet werden können, um den Selektionsbildschirm vorzubelegen.

3. Starten Sie die Transaktion FIBF, um ein Kundenprodukt anzulegen und den BTEs 00001070 und 00001080 die gerade angelegten Funktionsbausteine zuzuordnen.
4. Wählen Sie im Menü **EINSTELLUNGEN • PRODUKTE • ...EINES KUNDEN**.
5. Legen Sie ein neues Kundenprodukt ZFIDOC mit der Beschreibung »Erweiterung Belegerfassung« an, wie in Abbildung 5.2 zu sehen ist, und aktivieren Sie es.



Abbildung 5.2 Kundenprodukt für BTEs 00001070 und 00001080

6. Kehren Sie zurück in das Startbild der Transaktion FIBF, und wählen Sie im Menü **EINSTELLUNGEN • P/S-BAUSTEINE • ...EINES KUNDEN**.
7. Erstellen Sie neue Einträge für die BTEs 00001070 und 00001080, und ordnen Sie die Funktionsbausteine Z_SAMPLE_INTERFACE_00001070 und Z_SAMPLE_INTERFACE_00001080 zu (siehe Abbildung 5.3).

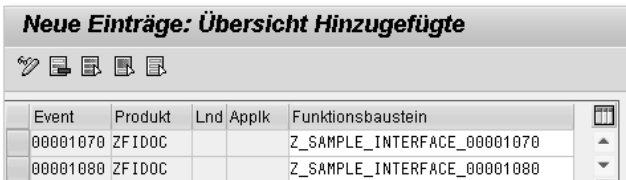


Abbildung 5.3 Zuordnung der BTEs zu Funktionsbausteinen

Damit ist der neue Funktionscode aktiviert, und der Aufruf der OP-Liste wurde zugeordnet. Bei der Erfassung der Belegposition in einer klassischen Buchungstransaktion sehen Sie nun wie in Abbildung 5.4 einen zusätzlichen Menüeintrag und eine zusätzliche Drucktaste mit dem Funktionstext OPS ANZEIGEN. Abhängig von der bearbeiteten Belegzeile, wird eine Sachkonten-, Debitoren- oder Kreditoren-OP-Liste zum aktuellen Konto angezeigt.

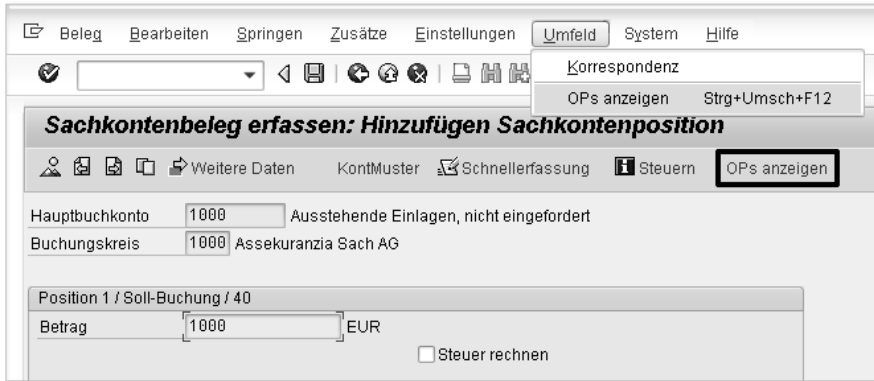


Abbildung 5.4 Zusätzlicher Menüeintrag und zusätzliche Drucktaste in klassischer Buchungstransaktion

Auch in Enjoy-Transaktionen werden der zusätzliche Menüeintrag und die zusätzliche Drucktaste angezeigt und beziehen sich dort auf die letzte Zeile der Positionstabelle bzw. auf den erfassten Debitor oder Kreditor, falls die Positionstabelle leer ist. In Abbildung 5.5 ist das beispielhaft für Transaktion FB60 (Kreditorenrechnung erfassen) dargestellt.

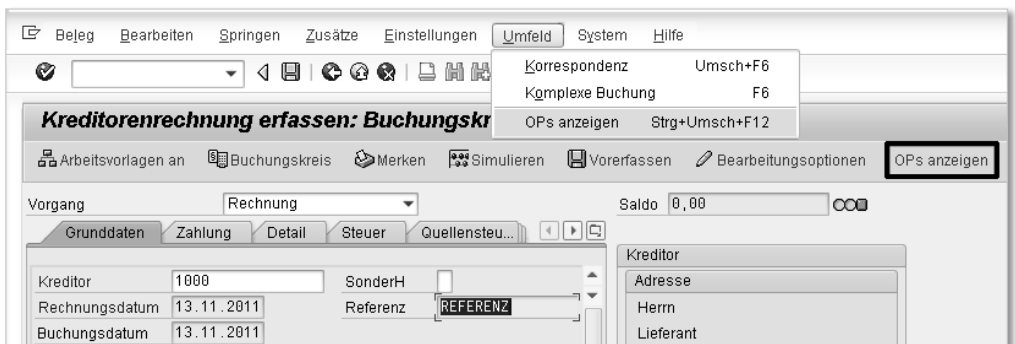


Abbildung 5.5 Zusätzlicher Menüeintrag und zusätzliche Drucktaste in Enjoy-Transaktion

Nachdem Sie nun gesehen haben, wie Sie einen zusätzlichen Funktionscode anlegen, geht es nun weiter mit der Pflege zusätzlicher Belegkopfdaten.

5.3.2 Pflege zusätzlicher Belegkopfdaten in Enjoy-Transaktion für Sachkontenbelege

Mit dem BAdI FI_HEADER_SUB_1300 existiert eine Erweiterung, mit der in den Grunddaten der Enjoy-Transaktion zur Erfassung von Sachkontenbelegen ein kundeneigener Subscreen eingeblendet werden kann. Dieser Subscreen kann auch zur Erfassung zusätzlicher Belegkopfinformationen verwendet werden. In diesem Abschnitt soll nun beispielhaft ein Subscreen definiert werden, über den ein zusätzliches Feld aus einer Append-Struktur zu Tabelle BKPF gepflegt werden kann.

In Anlehnung an die Beispielimplementierung zum BAdI in der zugehörigen SAP-Dokumentation unterscheidet sich diese Erweiterung technisch etwas von der Erweiterungen der Stammdatenpflegedialoge, die Sie in Teil 1, »Kundeneigene Erweiterungen von Stammdaten«, kennengelernt haben. Dort wurde der Subscreen in einem Modulpool definiert, und aus den Methoden der BAdI-Implementierung wurden verschiedene Forms im Modulpool aufgerufen, um Daten zwischen Standardlogik und kundeneigenem Subscreen auszutauschen.

Im nun folgenden Beispiel erfolgt die Definition des kundeneigenen Screens in einer Funktionsgruppe, und die BAdI-Implementierung dient als *Proxy-Klasse* für den Datenaustausch zwischen Standardlogik und kundeneigenem Subscreen. Das bedeutet, dass sowohl die Standardlogik als auch die Ablauflogik des Screens BAdI-Methoden verwenden, um Daten mit der BAdI-Implementierung auszutauschen. Die BAdI-Implementierung selbst hält demnach den aktuellen Stand der bearbeiteten Daten und gibt ihn bei Bedarf zur Pflege an den Subscreen oder zur weiteren Verarbeitung an die Standardlogik zurück.



Abbildung 5.6 Append-Struktur ZBKPF zu Tabelle BKPF

Im Beispiel soll ein Textfeld für eine kundeneigene Referenz in den Belegkopf aufgenommen werden. Definieren Sie deshalb, wie in Abbildung 5.6 zu sehen ist, eine Append-Struktur ZBKPF zu Tabelle BKPF mit dem 40-stelligen kundeneigenen Feld ZZXBLENR. Anschließend definieren Sie einen Subscreen für die Pflege des zusätzlichen Feldes und ordnen es dem BAdI FI_HEADER_SUB_1300 zu. Gehen Sie dazu wie folgt vor:

1. Legen Sie eine Funktionsgruppe ZFI_HEADER_SUB_1300 an, und definieren Sie im TOP-Include LZFI_HEADER_SUB_1300TOP einen Tabellenarbeitsbereich BKPF.
2. Legen Sie in der Funktionsgruppe ZFI_HEADER_SUB_1300 ein Dynpro 1300 an, wählen Sie SUBSCREEN als DYNPROTYP, und platzieren Sie das Feld ZZXBLENR aus dem Tabellenarbeitsbereich BKPF darauf (ausführlicher wurde dies in Abschnitt 2.4.1, »Subscreen für das Zusatzfeld anlegen«, beschrieben).
3. Legen Sie eine Implementierung ZFI_HEADER_SUB_1300 zum BAdI FI_HEADER_SUB_1300 an. Das BAdI ist filterabhängig und verwendet als Filter das Land des Buchungskreises, in dem der Beleg erfasst wird. Verwenden Sie hier das Land DE.
4. Wechseln Sie auf den Karteireiter SUBSCREENS, und ordnen Sie das Dynpro 1300 aus dem Rahmenprogramm SAPLZFI_HEADER_SUB_1300 der Funktionsgruppe ZFI_HEADER_SUB_1300 zu.
5. Speichern und aktivieren Sie die BAdI-Implementierung.

Sachkontenbeleg erfassen: Buchungskreis 1000

Arbeitsvorlagen an | Buchungskreis | Merken | Simulieren | Vorerfassen | Bearbeitungsoptionen

Grunddaten | Detail

Belegdatum	<input type="text"/>	Währung	EUR
Buchungsdatum	13.01.2012		
Referenz	<input type="text"/>		
Belegkopftext	<input type="text"/>		
Übergreifd.Nr	<input type="text"/>		
Buchungskreis	1000	Assekuranzia Sach AG München	
Kunden-Referenz	<input type="text"/>		

Betragsinformation

Summe Soll: EUR

Summe Haben: EUR

Abbildung 5.7 Transaktion FB50 mit kundeneigenem Subscreen

Wenn Sie nun die Transaktion FB50 starten, sehen Sie Ihr gerade angelegtes zusätzliches Feld auf dem Karteireiter GRUNDDATEN (siehe Abbildung 5.7). Bisher ist das Feld allerdings noch ohne Funktion. Damit der eingegebene

Wert des Feldes auch in der Tabelle BKPF gespeichert wird, müssen Sie noch den Datenfluss zwischen Standardlogik und kundeneigenem Subscreen implementieren. Dazu existieren die folgenden vier Methoden:

► **Methode PUT_DATA_TO_SCREEN_PBO**

Die Methode `PUT_DATA_TO_SCREEN_PBO` wird in der PBO-Logik des GRUNDDATEN-Dynpros vor der PBO-Logik des kundeneigenen Dynpros durchlaufen. Sie übergibt die aktuellen Belegkopfdaten und ein Flag (`IM_X_NO_INPUT`) an die BAdI-Implementierung, das anzeigt, ob das Buchungsdatum eingabebereit ist.



Flag »Buchungsdatum eingabebereit«

Das Flag wird an dieser Stelle übergeben, weil die Erweiterung auch zur Bestimmung eines abweichenden Buchungsdatums verwendet werden kann. In diesem Beispiel soll das kundeneigene Referenzfeld nur eingabebereit sein, wenn auch das Buchungsdatum eingabebereit ist.

► **Methode GET_DATA_FROM_SCREEN_PBO**

Die Methode `GET_DATA_FROM_SCREEN_PBO` wird im Gegensatz zu den anderen Methoden nicht im Standardprogramm `SAPMF05A` aufgerufen, sondern aus dem kundeneigenen Subscreen. Sie wird in diesem Beispiel dazu verwendet, in der PBO-Logik des kundeneigenen Screens Daten aus der BAdI-Implementierung abzufragen.

► **Methode PUT_DATA_TO_SCREEN_PAI**

Die Methode `PUT_DATA_TO_SCREEN_PAI` wird in der PAI-Logik des GRUNDDATEN-Dynpros aufgerufen und übergibt die bearbeiteten Belegkopfdaten an die BAdI-Implementierung, bevor die PAI-Logik des kundeneigenen Screens ausgeführt wird.

► **Methode GET_DATA_FROM_SCREEN_PAI**

Die Methode `GET_DATA_FROM_SCREEN_PAI` wird ebenfalls in der PAI-Logik des GRUNDDATEN-Dynpros aufgerufen und fragt nach der Verarbeitung der PAI-Logik des kundeneigenen Screens die bearbeiteten Daten aus der BAdI-Implementierung ab.

Wie schon erwähnt, soll in dieser Erweiterung die BAdI-Implementierung nur als Proxy-Klasse für den Austausch von Daten zwischen Standardlogik und kundeneigenem Subscreen dienen. Das Interface `IF_EX_FI_HEADER_SUB_1300` enthält bereits Attribute zu den Daten, die über die genannten vier Methoden ausgetauscht werden:

- ▶ das Attribut `BKPF` für Belegkopfdaten
- ▶ das Attribut `X_NO_INPUT` für das Flag, das abfragt, ob das Feld `BUCHUNGSDATUM` eingabebereit ist

Die Implementierung der Methoden des BAdIs `FI_HEADER_SUB_1300` ist daher denkbar einfach und in Listing 5.3 zu sehen.

```
METHOD if_ex_fi_header_sub_1300~put_data_to_screen_pbo.
    me->if_ex_fi_header_sub_1300~bkpf = im_bkpf.
    me->if_ex_fi_header_sub_1300~x_no_input
        = im_x_no_input.
ENDMETHOD.
METHOD if_ex_fi_header_sub_1300~get_data_from_screen_pbo.
    ex_bkpf = me->if_ex_fi_header_sub_1300~bkpf.
    ex_x_no_input
        = me->if_ex_fi_header_sub_1300~x_no_input.
ENDMETHOD.
METHOD if_ex_fi_header_sub_1300~put_data_to_screen_pai.
    me->if_ex_fi_header_sub_1300~bkpf = im_bkpf.
ENDMETHOD.
METHOD if_ex_fi_header_sub_1300~get_data_from_screen_pai.
    ex_bkpf = me->if_ex_fi_header_sub_1300~bkpf.
ENDMETHOD.
```

Listing 5.3 Methoden des BAdIs `FI_HEADER_SUB_1300`

In allen vier Methoden werden die Parameter in den Attributen des Interface `IF_EX_FI_HEADER_SUB_1300` gespeichert bzw. aus den Attributen zurückgegeben. Die BAdI-Implementierung bietet daher ausschließlich einen Zugriff auf die Attribute `BKPF` für die Standardlogik und `X_NO_INPUT` für den kundeneigenen Subscreen.

Der Datenaustausch zwischen Standardlogik und BAdI-Implementierung erfolgt, wie beschrieben wurde: Vor der Ausführung der PBO- und der PAI-Logik des kundeneigenen Screens werden die relevanten Daten an die BAdI-Implementierung übergeben. Nach der Ausführung der PAI-Logik des kundeneigenen Screens werden die Daten wieder aus der BAdI-Implementierung abgefragt. Für eine korrekte Funktionsweise der Erweiterung müssen Sie nun noch in der PBO-Logik des Screens die Datenabfrage aus der BAdI-Implementierung und in der PAI-Logik die Rückgabe der bearbeiteten Daten implementieren.

Gehen Sie dazu wie folgt vor:

1. Definieren Sie im TOP-Include der Funktionsgruppe ZFI_HEADER_SUB_1300 eine globale Variable GOBJ_EXIT vom Typ IF_EX_FI_HEADER_SUB_1300 für den Zugriff auf die aktive BAdI-Implementierung.
2. Fügen Sie im Dynpro 1300 ein PBO-Modul STATUS_1300 hinzu, und fügen Sie den Quellcode aus Listing 5.4 ein.

```

MODULE status_1300 OUTPUT.
  DATA: lv_x_no_input TYPE c.
  * BAdI-Implementierung instanzieren
  IF gobj_exit IS INITIAL.
    CALL METHOD
      cl_exithandler=>get_instance_for_subscreens
      CHANGING
        instance = gobj_exit
    EXCEPTIONS
      OTHERS = 6.
  IF sy-subrc <> 0.
    MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno
      WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.
  ENDIF.
ENDIF.

* Eingabebereitschaft abfragen
CALL METHOD gobj_exit->get_data_from_screen_pbo
  EXPORTING
    flt_val = 'DE'
  IMPORTING
    ex_x_no_input = lv_x_no_input.

* Eingabebereitschaft setzen
LOOP AT SCREEN.
  IF lv_x_no_input = 'X'.
    screen-input = 0.
    MODIFY SCREEN.
  ENDIF.
ENDLOOP.
ENDMODULE. " STATUS_1300 OUTPUT

```

Listing 5.4 PBO-Modul STATUS_1300 des kundeneigenen Subscreens

Im Quellcode wird zunächst mithilfe der Klasse CL_EXITHANDLER die aktive Implementierung des BAdIs FI_HEADER_SUB_1300 abgefragt. Die aktive Implementierung wurde vorher bereits von der Standardlogik gesetzt,

sodass sowohl die Standardlogik als auch der Subscreen auf dieselbe Instanz der BAdI-Implementierung zugreifen.

Danach werden die aktuellen Daten aus der BAdI-Implementierung abgefragt, die vor der Ausführung der PBO-Logik des Screens vom Standardprogramm gesetzt wurden. Da anschließend nur das Flag für die Steuerung der Eingabebereitschaft benötigt wird, wird auch nur dieses abgefragt. Der Wert des Feldes ZZXBLNR wird außerhalb des Screens nicht verändert und braucht deshalb nicht aus der BAdI-Implementierung gelesen werden.

3. Fügen Sie als Nächstes ein PAI-Modul USER_COMMAND_1300 mit dem Quellcode aus Listing 5.5 ein.

```
MODULE user_command_1300 INPUT.
  DATA: lv_bkpf_changed TYPE bkpf.
  * Aktuelle Daten abfragen
  CALL METHOD gobj_exit->get_data_from_screen_pai
    EXPORTING
      flt_val = 'DE'
    IMPORTING
      ex_bkpf = lv_bkpf_changed.
  lv_bkpf_changed-zzxblnr = bkpf-zzxblnr.

  * Bearbeitete Daten zurückgeben
  CALL METHOD gobj_exit->put_data_to_screen_pai
    EXPORTING
      im_bkpf = lv_bkpf_changed
      flt_val = 'DE'.
ENDMODULE.                " USER_COMMAND_1300  INPUT
```

Listing 5.5 PAI-Modul USER_COMMAND_1300 des kundeneigenen Screens

Im Quellcode wird als Erstes der aktuelle Zustand der Belegkopfdaten aus der BAdI-Implementierung gelesen. So enthält die lokale Variable den aktuellen Wert aller Felder bis auf das Feld ZZXBLNR. Dieses Feld wird anschließend aus dem Tabellenarbeitsbereich gesetzt, das heißt, es entspricht danach dem Wert, der im Subscreen eingegeben wurde. Zum Schluss werden die vollständig bearbeiteten Belegkopfdaten zurück in die BAdI-Implementierung geschrieben, wo sie wieder von der Standardlogik abgefragt werden können.

4. Sichern und aktivieren Sie die Funktionsgruppe ZFI_HEADER_SUB_1300.

Die Erweiterung ist nun so weit fertiggestellt, dass das zusätzliche Feld über den kundeneigenen Subscreen gepflegt werden kann und der bearbeitete Wert an die Standardlogik zurückgegeben wird. Wenn Sie sich allerdings die

Aufrufstelle der Methode `GET_DATA_FROM_SCREEN_PAI` in der PAI-Logik des GRUNDDATEN-Dynpros (Form `D0100_POST_APPL_SUB` in Programm `SAPMF05A`) anschauen, sehen Sie, dass nur die Felder `BUDAT`, `XREF1_HD` und `XREF2_HD` aus der zurückgegebenen Struktur übernommen werden. Hier können Sie allerdings eine einfache implizite Erweiterung vor die `ENDFORM`-Anweisung einfügen, die Sie in Listing 5.6 sehen.

```
ENHANCEMENT 4 Z_SAPMF05A_FI_HEADER_SUB_1300.
  IF g_appl_sub_active = char_x.
    bkpf-zzxblnr = ld_bkpf-zzxblnr.
  ENDIF.
ENDENHANCEMENT.
```

Listing 5.6 Implizite Erweiterung für die Übernahme des zusätzlichen Feldes

In dieser Erweiterung wird das Feld `ZZXBLNR` aus der zurückgegebenen Struktur übernommen. Da eine implizite Erweiterung nur nach der letzten Zeile einer Form eingefügt werden kann, muss hier noch einmal die Abfrage über die Variable `G_APPL_SUB_ACTIVE` wiederholt werden, ob die Erweiterung aktiv ist.



Implizite Erweiterung zur Rückgabe des Feldes `ZZXBLNR`

Es ist gut möglich, dass in zukünftigen SAP ERP-Releases diese implizite Erweiterung nicht mehr notwendig sein wird, weil alle Daten aus dem BAdI über die `MOVE-CORRESPONDING`-Anweisung übernommen werden, wie es heute schon für die Erweiterung der Enjoy-Transaktionen zur Rechnungserfassung der Fall ist (siehe Abschnitt 5.3.3).

Wenn Sie anschließend in der Transaktion `FB50` einen Beleg erfassen und das Feld `KUNDEN-REFERENZ` füllen, können Sie danach in der Tabelle `BKPF` sehen, dass der eingegebene Wert mit gespeichert wird. Beachten Sie dabei auch, dass, sobald das Buchungsdatum nicht mehr eingabebereit ist, auch in das Feld `KUNDEN-REFERENZ` nichts mehr eingegeben werden kann.

5.3.3 Pflege zusätzlicher Belegkopfdaten in Enjoy-Transaktion für Debitoren- und Kreditorenbelege

Ähnlich wie die Enjoy-Transaktion für die Erfassung von Sachkontenbelegen können auch die Enjoy-Transaktionen für die Erfassung von Debitoren- und Kreditorenbelegen um kundeneigene Subscreens erweitert werden. Neben den Belegkopfinformationen können dabei auch die Positionsinformationen zur erfassten Debitoren- bzw. Kreditorenposition bearbeitet werden. Hierfür

Index

A

Accruel Engine 340, 358
Akontobuchung 205
Aktivität 100, 124
Änderungsbeleg 57, 127
 anzeigen 131
 Feldänderung 131
 Fortschreibung 130
 Verbucher generieren 129
Änderungsbelegobjekt 128
Append-Struktur 64, 87, 152
 BKPF 160
 BSAS 165
 BSEG 160
 BSIS 165
 INVFO 160
 KNB1 87
 RFPOS 32
 SKB1 64
Applikationskennzeichen,
 kundeneigenes 228
Arbeitsgebiet
 GBLR 172
 GBLS 172
Archivierung
 Abgrenzungsobjekt 357
 Bankenstammdaten 358
 Buchhaltungbeleg 356
 Debitorenstammdaten 354
 Kreditorenstammdaten 355
A-Segment 53
Ausgleich 143
Auslieferungsklasse
 E 74
 S 73
AWV-Meldung 352

B

BAdI 38
 BADI_FDCB_SUBBAS01 159
 CUSTOMER_ADD_DATA 91
 CUSTOMER_ADD_DATA_BI 109

BAdI (Forts.)
 CUSTOMER_ADD_DATA_CS 91
 FAGL_ITEMS_CH_DATA 165, 285
 FAGL_ITEMS_MENUE01 39, 285
 FAGL_ITEMS_MENUE02 285
 FEB_BADI 193
 FI_F110_SCHEDULE_JOB 241
 FI_HEADER_SUB_1300 152
 FI_ITEMS_MENUE01 39, 276
 FI_ITEMS_MENUE02 276
 FIEB_CHANGE_BS_DATA 190
 FIEB_CHANGE_STATEMNT 191
 filterabhängiges 39, 153, 191
 finden 44
 kernelbasiertes 38, 45
 VENDOR_ADD_DATA 45, 116, 119
 VENDOR_ADD_DATA_BI 135
 VENDOR_ADD_DATA_CS 116, 119,
 131
BAdI, klassisches 38
 aktivieren 43
 anzeigen 39
 Definition 38
 filterabhängiges 39
 Filtertyp 40
 Funktionscode 40, 276
 Implementierung 38, 40
 Interface 40
 mehrfach nutzbares 39
 Methode 40, 42
 Subscreen 40
BAdI, neues 38, 45, 47
 Aktivierung 49
 Erweiterungsimplementierung 47
 implementierende Klasse 48
 Interface 46
 Methode 46
BAdI-Methode → Methode
Bankenfindung 243
Bankenstammdaten 297, 358
Bankverbindung 296
Bankverrechnungskonto 181
Batch-Input 168
Batch-Input-Mappe 105, 168

Batch-Input-Programm 105, 108, 135, 167
 Batch-Input-Recorder 108
 Batch-Input-Struktur 105
 Erweiterung 107
 kundeneigene 135
 Batch-Input-Verfahren 105, 106
 Bearbeitungsmodus
 Debitorenstammdaten 98
 Kreditorenstammdaten 121
 Sachkontenstammdaten 76
 Begleitzettel 256
 Beleg 139, 311
 Belegaufteilung 310
 Belegbuchung, Erweiterung 166
 Belegerfassung 301
 zusätzlicher Funktionscode 147
 zusätzliches Belegkopffeld 152, 158
 Belegjournal 339
 Belegkopf 139
 Belegposition 140
 Belegvorerfassung 314
 Belegzeile, zusätzliche erzeugen 142, 166
 Bestandskonto 53, 80
 Bilanz 144, 161, 339, 348
 Bildgruppe 88, 90, 91, 118, 134
 B-Segment 53
 BTE → Business Transaction Event
 Buchhaltungsbeleg 139, 237, 357
 Buchhaltungssicht 86, 114
 Buchung
 automatische 167
 buchungskreisübergreifende 142, 166
 manuelle 167
 Buchungsbereich 181
 Buchungsdialo g 147
 Buchungskreis 139, 145, 218
 Buchungskreissicht 96, 121
 Buchungsregel 181
 Buchungsschlüssel 140
 Buchungsschnittstelle 168, 193, 195
 Buchungszeile 140
 Business Add-In → BAdI
 Business Transaction Event 27
 aktive Komponente 31
 Aktivierung 29, 34
 Alternativprozess 36
 anzeigen 30

Business Transaction Event (Forts.)
 Applikationskennzeichen 29, 34, 36, 227
 finden 36
 Gruppierung 29
 Implementierung 32
 Konfiguration 28, 34
 Kunde 30
 länderspezifisches 35, 36
 Musterfunktionsbaustein 31
 Partner 29
 Partneridentifikation 29
 Produkt 30, 34
 RFC-Destination 29
 SAP-Anwendung 29
 Typ 28
 BZÜ-Verfahren 202

C

CALL CUSTOMER-FUNCTION 23, 26
 Call Transaction 105, 168
 Cluster-Tabelle 141
 CO-FI-Echtzeitintegration 318
 CPD-Daten 142
 Customizing-Include 64
 CI_BKNB1 107
 CI_COBL 162
 CI_CSXS 20
 CI_FAGLPOXS 286

D

Datenelement 56
 Datenmodell, Kreditorenstammdaten 113
 Datenträgerverwaltung 331
 Dauerbeleg 318
 Debitoren-Informationssystem 339
 Debitorensaldo 338
 Debitorenstammdaten 85, 86, 289, 355
 Dict-/Programmfelderfenster 70
 Direct Input 168, 170
 Dirty-Assign 78
 DME-Engine 299
 Domäne, Wertebereich 57
 Domänenfestwert 20, 221

Dynpro

- SAPLKMA1 0399 20
- SAPLKMA1 3399 20
- SAPLXKM1 0999 20
- SAPMF02D 0110 108
- SAPMF02K 4000 134
- SAPMF05A 0300 147
- Typ 70, 89, 118, 153
- ZKN_ERWEITERUNG 0100 89
- ZLF_ERWEITERUNG 0100 118
- ZSK_ERWEITERUNG 0100 70

Dynpro-Feld bestimmen 132

E

EDI

- Avis 364
- EDI-Referenz-IDoc 365
- Eingangsrechnung 363
- Kontoauszug 365
- Zahlung 364

Einbildtransaktion 146

Eingabehilfe/-prüfung 65

Einkaufssicht 114, 121

Einzelpostenliste 267, 334

ALV-Grid 268

ALV-List 268

Anzeigevariante 269

Erweiterung 164, 269, 284

Kreditor 35

Menüerweiterung 39, 273, 285

Sachkonto 43

Vorselektion 281, 286

zusätzlicher Funktionscode 273

zusätzliches Feld 280, 285

Elektronischer Kontoauszug 179, 319

Ausgleich Personenkonto 182, 208

Buchungsphase 182, 192

Buchungsregel 208

Interpretationsphase 180, 188

Kontenfindung 180, 193

Kontoauszugsdaten einlesen 179, 183

kundeneigene Buchung 192

Nachbearbeitung 182, 195

Standardbuchung deaktivieren 189

Standardbuchung erweitern 193, 194

Suchmuster 210

Umsatzregistrierung 188

Elektronischer Kontoauszug (Forts.)

Verwendungszweck aufbereiten 211

Enjoy-Transaktion 146

Erfolgskonto 53

Eröffnungsbilanz 144

Erweitertes Zahlprogramm 332

Erweiterung, implizite 49, 158

Erweiterungsmöglichkeit, technische 15

Erweiterungspunkt

expliziter 45

impliziter 45

Erweiterungsspot 45

CUSTOMER_ADD_DATA 92

CUSTOMER_ADD_DATA_BI 109

CUSTOMER_ADD_DATA_CS 94

VENDOR_ADD_DATA 119

VENDOR_ADD_DATA_CS 120

Erweiterungstechnik 15

Debitorenstammdaten 87

Kreditorenstammdaten 115

F

Feldänderung 133

Feldstatus 54, 140

Feldstatusgruppe 162

FI-Bewertungslauf 343

filterabhängig 96, 121

Filterwert 96

Funktionsbaustein

AC_DOCUMENT_CREATE 170

AC_DOCUMENT_DIRECT_INPUT 170

AC_DOCUMENT_POST 170

BF_FUNCTIONS_READ 37

DATE_TO_PERIOD_CONVERT 145

FAGL_ITEMS_DISPLAY 268

FI_PAYM_ACC_SHEET_PRINT 256

FI_PAYMEDIUM_SAMPLE 254

FI_PAYMEDIUM_SAMPLE_DETAILS
260

FI_PRINT_DUNNING_NOTICE 236

FI_PRINT_DUNNING_NOTICE_SMARTF
235

FIEB_028_ALGORITHM 205

FIEB_EXTRACT_MINUS_NUMBERS
207

FIEB_EXTRACT_NUMBERS 207

GENERATE_DUNNING_DATA 219

Funktionsbaustein (Forts.)
GL_ACCT_MASTER_MAINTAIN 55,
 66, 71
GL_ACCT_MASTER_SAVE 67
MODX_FUNCTION_ACTIVE_CHECK 26
PC_FUNCTIONS_READ 37
POSTING_INTERFACE_CLEARING 169
POSTING_INTERFACE_DOCUMENT
 169
POSTING_INTERFACE_END 168
POSTING_INTERFACE_RESET_CLEAR
 169
POSTING_INTERFACE_REVERSE_DOC
 170
POSTING_INTERFACE_START 168
SAMPLE_INTERFACE_[BTE] 31
SAMPLE_PROCESS_[BTE] 31
TABSTRIP_INIT 72
TABSTRIP_LAYOUT_READ 72
*Z_FEB_1** 194
*Z_FEB_2** 194
Z_FIEB_XXX_ALGORITHM 205

Funktionscode 147

Funktions-Exit
EXIT_RFEKA400_001 186
EXIT_SAPLKMA1_001 23
EXIT_SAPLKMA1_002 23
EXIT_SAPMF02D_001 103
finden 26

Funktionsgruppe
anlegen 129
ATAB 72
FAGL_ITEMS_SELECT 268
FIPI 168
GL_ACCOUNT_MASTER_MAINTAIN 66
KMA1 20
RWCL 167
RWFI 170
XKM1 20

G

Geschäftsvorfallcode 180
 GET BADI 49
 GET CURSOR FIELD 132
 GET_EXIT_TITLES 173
 Gewinn- und Verlustrechnung → GuV
 Granularität 253, 263

Gruppenrahmen 72
 GuV 144, 161, 339, 347

H

Hauptbuch
Allokation 348
statistische Kennzahlen 318
 Hauptbuchhaltung, Planung 348
 Hauswährung
Betrag 140
parallel geführte 145

I

Include
LGL_ACCOUNT_MASTER_MAINTAINFFR
 66
ZXF01U06 186
ZXKM1U01 24
ZXKM1U02 24

Intercompany-Abstimmung 344
 Interface, IF_EX_FI_HEADER_SUB_1300
 154

Interpretationsalgorithmus 188, 200
Ausgangsscheck 201
Ausgleichsinformation 207
Ausgleichsvorgang 202
Belegnummernsuche 202
kundeneigener 205, 207
Nummernbereich 201, 207
Referenzbelegnummernsuche 202
Standardalgorithmus 201
Verwendungszweck 204, 206

K

Klasse
CL_EXITHANDLER 44, 156, 161
CL_FEBAN_ALV_GRID 199
CL_FEBAN_SIMPLE_TREE 199
 Klassisches BAdI → BAdI, klassisches
 Kontenplan 53, 72, 75
 Kontierung 308

Kontierungsblock 161
Expertenmodus 163
Light-Modus 163
 Kontierungsfeld 161
 Kontierungskombination, Validierung
 317
 Konto 141
Einzelpostenführung 143
OP-Führung 143
 Kontoart 140, 166
 Kontoauszug, Rohdaten 183
 Kontoauszugsnachbearbeitung 182, 195
Automatisierung 197
Baumdarstellung 198
Listendarstellung 198
Teilfenster 199
Umsatzregistrierung 198
 Konzernwährung 145
 Korrespondenz 347
 Kreditoren-Informationssystem 339
 Kreditorenstammdaten 113, 292, 356
Erweiterungsmöglichkeit 114
Zusatzfeld 114
zusätzliche Prüfung 133
 Kundeneigener Subscreen
Kreditorenstammdaten 118
Sachkontenstammdaten 69

L

Logische Datenbank
DDF 268
KDF 268
SDF 267

M

Mahnbereich 218
 Mahnbestand 218, 230
 Mahndaten 217
 Mahndruck 218, 234
 Mahngebühr 217
 Mahnlauf 217, 324
Ausführungsdatum 217
Druckphase 218
Einzelpostenregistrierung 228

Mahnlauf (Forts.)
gesperrtes Konto 218
Gruppierung 220
kundeneigener Selektionsbildschirm 221
Laufidentifikation 217
Nachbearbeitung 220
Selektionsphase 217
Zahlungsvorschlag 238
zusätzliches Selektionskriterium 221,
 225
 Mahnschreiben 218, 235
 Mahnselektion 217
Erweiterung 226
kundeneigene Gruppierung 229
 Mahnverfahren 217, 234
 Manueller Kontoauszug 324
 Mapping 211
 Mehrwertsteuer, Pro-rata-Korrektur 352
 Menü-Exit 22
 Merkmal 269
 Methode 41
CHANGE_DATA 190
CHANGE_ITEM 285
CHANGE_POSTING_DATA 193
CHECK_ADD_ON_ACTIVE 92, 120
CHECK_ALL_DATA 133
CHECK_DATA_CHANGED 102, 122,
 126
CHECK_DATA_ROW 136
CHECK_PARAMETER 241
FILL_FT_TABLE_USING_DATA_ROWS
 109, 136
GET_CHANGEDOCS_FOR_OWN_TABLES
 131
GET_DATA 98, 101
GET_DATA_CHANGED 98
GET_DATA_FROM_SCREEN 158
GET_DATA_FROM_SCREEN_OBJECT
 160
GET_DATA_FROM_SCREEN_PBO 154
GET_FIELDNAME_FOR_CHANGEDOC
 131
GET_TAXI_SCREEN 94, 120
INITIALIZE_ADD_ON_DATA 122
LIST_ITEMS01 41, 278
PUT_DATA_TO_SCREEN_OBJECT 159
PUT_DATA_TO_SCREEN_PA1 154
PUT_DATA_TO_SCREEN_PBO 154
READ_ADD_ON_DATA 122, 125

Methode (Forts.)

SAVE_DATA 122, 126
SET_DATA 98, 99, 122
SHOW_BUTTONS 277

Modifikationsgruppe 76

Modulpool 124, 132, 172

MT940 179, 183

Erweiterung 185

Feld 184

Satz 183

Muster 211

Musterbeleg 318

N

Nachrichtentyp

ACCONF 362

ACLPAY 362

ACLREC 363

ACPJMM 362

FIDCC1 359, 360

FIDCC2 359, 360

FIDCCH 359

FIDCMT 359

FIPAYM 362

FIROLL 359

GLMAST 358

INVOIC 361

Nebenbuch 146

Neues Hauptbuch 144

New General Ledger 144

New GL 144

O

OP-Ausgleich 298

Open FI 27

Open-FI-Erweiterung 147

OP-Führung 143

P

P&S-Schnittstelle 28

00001070 148

00001080 148

P&S-Schnittstelle (Forts.)

00001610 274

00001620 273

00001630 281

00001640 271

00001650 30, 164, 281

00001719 235

00001720 235

00001750 222, 223

00001761 227, 229

00002110 251

00002310 81

00002810 188

00002830 197

00002840 197

00002850 198

PAI-Modul

Belegerfassung 154, 161

Kreditorenstammdatenpflege 132

Sachkontenstammdatenpflege 79

Parameter, formatspezifischer 254, 255,
261, 262

Payment Medium Workbench (PMW)

239, 253

Pre-Service 255

Verwendungszweck 257

PBO-Modul

Belegerfassung 154, 161

Debitorenstammdatenpflege 100

Kreditorenstammdatenpflege 124

Sachkontenstammdatenpflege 76, 79

Periode 145

Periodenwert 145

Periodische Arbeiten 179

Pflagedialog

Debitorenstammdaten 88

Kreditorenstammdaten 117

Sachkontenstammdaten 68

Pflege-View 62

Berechtigungsgruppe 63

Beziehung 62

VC_TAMLAYA_00 72

View-Feld 62

PMW → Payment Medium Workbench
(PMW)

Positionsinformation 158

Posten

offen/ausgeglichen 143

offener 238

Posten- und Saldenverzinsung 341
 Programm
DMEECONVERT1 299
FAGL_ACCOUNT_ITEMS_GL 164, 267
FAGL_PLAN_COPY 348
RFAWVZ40N 352
RFBIBL00 167
RFBIDE00 105
RFBIDE10 106
RFBIKR00 135
RFBIKR10 135
RFBISA00 83
RFDUZI00 341
RFDZIS00 341
RFDZIS00_PDF 342
RFEBBU00 183, 192
RFEBBU10 183, 188
RFEBKA00 183
RFEKA400 183
RFITEMAP 268
RFITEMAR 268
RFITEMGL 164, 267
*RFKORD** 347
RFKUZI00 341
RFKZIS00 341
RFKZIS00_PDF 342
RFMAHN21 220
RFPAYM_RESET 255
RFPOSEXTEND 165
RFPOSXEXTEND 35, 280
RFSZIS00 341
RFUMSV00 350
RFZALI20 240, 250
RGGBR000 172
RGGBS000 172
*SAPF056** 354
SAPF057 354
*SAPF058** 355
SAPF059 355
SAPF1100 240
SAPF110S 240
SAPF110V 240
SAPF130D 347
SAPF130K 347
SAPF150D2 219
SAPF150S2 219
SAPGL_ACCOUNT_MASTER_START 54
SAPLGUMD 163
SAPLKMA1 20

Programm (Forts.)
SAPLXKM1 20
SAPMF02D 86, 98
SAPMF02K 114
SAPMF05A 147, 167
SAPMF05L 147, 313
 Proxy-Klasse 152, 154
 Prozessschnittstelle 28
00001053 227
00001060 227
00001061 227, 230
00001810 244
00001820 246
00001830 247
00002810 192
00002820 189
00002830 195
00002850 197
00002870 198
 Prüfung 172
Debitorenstammdaten 103
Sachkontenstammdaten 80
 Publish&Subscribe-Schnittstelle →
 P&S-Schnittstelle

R

Referenz, externe/interne 257
 Referenzfeld 254, 257
 Regulierungsdaten 238
 Reporting 267
 Rückstellung 349

S

Sachkontengruppe 53, 71, 75
 Sachkontenstammdaten 53, 55, 288
 Saldo-Null-Prüfung 166
 Saldovortrag 144
 SAP Closing Cockpit 346
 SAP Enhancement Framework 45, 49
 SAP Smart Forms 235
 SAP-Erweiterung
aktivieren 25
anzeigen 16
Attribut 16

- SAP-Erweiterung (Forts.)
- COOMKS01* 16
 - Dokumentation* 16
 - Dynpro-Erweiterung* 20
 - erweiterte Suchhilfe* 26
 - Erweiterungsprojekt* 18
 - FEB00001* 190, 194
 - FEB00004* 183
 - finden* 26
 - Funktions-Exit* 23
 - GLX_MD01* 163
 - Implementierung* 18
 - Komponente* 15, 19
 - Menüerweiterung* 22
 - Menü-Exit* 22
 - Projektverwaltung* 18
 - SAPMF02D* 103
 - Screen-Exit* 21
 - Tabellenerweiterung* 20
 - Zuordnung* 19
- SAPscript-Formular 218
- Scheck, Einreichung 324
- Schlussbilanz 144
- Schnittstelle
- Buchhaltungsbeleg* 167
 - Debitorenstammdaten* 104
 - Kreditorenstammdaten* 135
 - Sachkontenstammdaten* 83
- Screen-Exit, SAPLXKM1 0999 21
- Sekundärindextabelle 142
- Sicht
- Debitorenstammdaten* 86, 96
 - Kreditorenstammdaten* 114, 121
- Skonto 238
- Soll-/Haben-Kennzeichen 140
- Sonderperiode 145
- SPA-/GPA-Parameter 150
- Special Ledger 162
- Stammdatenprüfung 163
- Stammdatentabelle 113, 116
- Standard-Interpretationsalgorithmus 200
- Steuer 309
- Steuerposition 142
- Steuerzeile 142
- Struktur
- BAPIRET2* 81
 - BBKPF* 168
 - BBSEG* 168
 - BBTAX* 168
- Struktur (Forts.)
- BBWITH* 168
 - BGR00* 105, 135
 - BKN00* 106
 - BKNA1* 105
 - BKNB1* 105
 - BKNBK* 106
 - BLF00* 135
 - BLFA1* 135
 - BLFB1* 135
 - BSELK* 168
 - BSELP* 168
 - CSKS_CI* 21
 - FAGLPOSE* 164
 - FAGLPO SX* 164, 286
 - FILITEXTS_AP* 270
 - FILITEXTS_AR* 270
 - FILITEXTS_GL* 270
 - FPAYH* 239
 - FPAYHX* 239
 - FPAYP* 239
 - FPM_PAYD* 260
 - FTCLEAR* 169
 - FTPOST* 169
 - FTTAX* 169
 - GLACCOUNT_CC CODE_DATA* 67
 - INVFO* 159
 - RFPOS* 32, 164
 - RFPOSX* 32, 164, 280
 - RFPOSXEXT* 35, 165, 280
- Subscreen 88, 152
- Belegkopffeld* 153
 - Debitorenstammdaten* 89
 - Kontoauszugsnachbearbeitung* 195
- Substitution 172
- Aktivierung* 176
 - Schritt* 172, 176
 - User-Exit* 171, 174
 - Zeitpunkt* 175
- Suchmuster 188, 210
- Anfangs- und Endzeichen* 212
 - Definition* 211
 - Mapping* 211
 - Muster* 211
 - Zielfeld* 213
 - Zuordnung* 213
- Summentabelle
- Hauptbuch* 144
 - Nebenbuch* 146

T

Tabelle

BKPF 139, 152
BSAD 143
BSAK 143
BSAS 143
BSEC 142
BSEG 140
BSEG_ADD 144
BSET 142
BSID 143
BSIK 143
BSIS 143
BVOR 142
CSKS 17
Datenart 59
FAGLBSAS 144
FAGLBSIS 144
FAGLFLEXT 144
FEBCL 180
FEBEP 180
FEBKO 180
FEBRE 180
Fremdschlüssel 60
GLT0 144
Größenkategorie 59
KNA1 85
KNB1 85
KNB5 85
KNBK 85
KNC1 146
LFA1 114
LFB1 114
LFBK 114
LFC1 146
LFM1 114
MAHNS 218
MAHNV 218
MHND 217
MHNC 217
REGUH 238
REGUP 238
REGUV 239
SKA1 53
SKAT 53
SKB1 53
T004 53, 72, 75
T021S 164, 280

Tabelle (Forts.)

T077S 53, 71, 75
TAMLAY0 72
TAMLAY1 72
TAMLAY2 72
TAMLAYA 72
TAMLAYB 72
TBE01 36

technische Einstellung 58
TPS01 36
TPS31 36
TPS32 36
TPS34 36
ZLFA1 116

Tabellenarbeitsbereich

KNB1 89
SKB1 70
ZLFA1 118, 123

Tabellen-Cluster, RFBGL 141

Tabellenpflegegenerator 63

Tabstrip 72

Tabstrip-Layout 74

Texttabelle 60

Textvariable, merkmalsbezogene 270

Transaktion 146

CMOD (Projektverwaltung SAP-Erweiterungen) 18, 25
F.01 (Bilanz) 339
F.11 (Hauptbuch aus Belegdatei) 340
F.30 (Debitoreninfosystem) 339
F.46 (Kreditoreninfosystem) 339
F.50 (Nachbelastung GuV) 347
F.56 (Dauerbelege löschen) 318
F.57 (Musterbelege löschen) 318
F.5D (Nachbelastung Bilanz) 348
F-03 (Sachkonto ausgleichen) 298
F104 (Rückstellung Forderung) 349
F107 (FI-Bewertungslauf) 343
F110 (Zahllauf) 239, 330
F111 (erweitertes Zahlprogramm, Zahlungsanforderungen) 237, 332
F150 (Mahnlauf) 219, 324
F-22 (Debitorenrechnung erfassen) 146
F-32 (Debitor ausgleichen) 298
F-44 (Kreditor ausgleichen) 298
F8BX (Online-Zahlung) 333
FAGLL03 (Sachkonten-Einzelposten) 43, 164, 267, 334
FB01 (Beleg buchen) 146, 169

Transaktion (Forts.)

- FB02 (Beleg ändern)* 147, 313
- FB03 (Beleg anzeigen)* 147, 313
- FB05 (Buchung mit Ausgleich)* 169
- FB08 (Beleg stornieren)* 170
- FB50 (Sachkontenbeleg erfassen)* 146, 153
- FB60 (Debitorenrechnung buchen)* 146
- FB70 (Kreditorenrechnung buchen)* 146
- FBF1 (Fremdwährungsbewertungen buchen)* 169
- FBL1N (Kreditoren-Einzelposten)* 35, 268, 334
- FBL3N (Sachkonten-Einzelposten)* 164, 267, 334
- FBL5N (Debitoren-Einzelposten)* 268, 334
- FBRA (Ausgleich zurücknehmen)* 169, 299
- FBTR* 354
- FBV* (Belegvorerfassung)* 317
- FD01 (Buchhaltungssicht, Anlage)* 86
- FD02 (Buchhaltungssicht, Änderung)* 86
- FD03 (Buchhaltungssicht, Anzeige)* 86
- FD10N (Debitoren-Saldenanzeige)* 338
- FDTA (Datenträgerverwaltung)* 331
- FEBAN (Kontoauszugsnachbearbeitung)* 183, 195, 198
- FF_5 (Kontoauszug einlesen)* 183, 201
- FF67 (Manueller Kontoauszug)* 324
- FF68 (Manueller Scheckeinreicher)* 324
- FIBF (BTE pflegen)* 28, 30, 222, 229, 252
- FIBLAPOP (Kreditorenzahlungsanordnung)* 333
- FIBLAROP (Debitorenzahlungsanordnung)* 333
- FIBLFFP (Free-Form-Zahlungsanordnung)* 333
- FINT (Postenverzinsung)* 342
- FK01 (Kreditorenstammdaten anlegen)* 114
- FK02 (Kreditorenstammdaten ändern)* 114
- FK03 (Kreditorenstammdaten anzeigen)* 114
- FRFT* 333

Transaktion (Forts.)

- FS00 (Einstellungen Sachkonto/Buchungskreis)* 54
 - FSPO (Einstellungen Sachkonto/Kontenplan)* 54
 - FSSO (Pflege Sachkontenstamm, Buchungskreis)* 54, 75
 - KALC (Kostenflüsse melden)* 319
 - KS02 (Einzelbearbeitung Kostenstelle)* 25
 - OB_GLACC21 (Konfiguration Sachkontenstamm)* 72, 74
 - OBBH (Substitutionen)* 175
 - S_RFID_PTVPRADPRC00 (Pro-rata-Korrekturen, Kalkulation)* 352
 - S_RFID_PTVPRADPRV00 (Pro-rata-Korrekturen, Variation)* 352
 - SCDO (Änderungsbeleg)* 128
 - SE11 (ABAP Dictionary)* 32
 - SE18 (BAdI-Definition)* 39, 45
 - SE19 (BAdI-Implementierung)* 40, 47, 277
 - SE37 (Function Builder)* 32
 - SHDB (Batch-Input-Recorder)* 108
 - SM30 (View pflegen)* 172
 - SM35 (Batch-Input-Mappe)* 105
 - SMOD (SAP-Erweiterungen)* 16, 26
 - XD02 (Debitor ändern)* 108
- Transaktionswährungsbetrag 141

U

- Umsatzsteuervergütung 354
- Umsatzsteuervoranmeldung 349
- User-Exit 15, 172

V

- Validierung
 - Schritt* 172
 - User-Exit* 171
- Variablentyp 269
- Verarbeitungslogik
 - Belegkopffeld* 153, 155, 159
 - Zusatzfeld* 76
 - Zusatzfeld für Debitorenstammdaten* 97

Verarbeitungslogik (Forts.)
Zusatzfeld Kreditorenstammdaten 121
 Verarbeitungsstruktur 65, 68
 Vertriebssicht 86, 96
 Verwaltungssatz 239
 Verwendungszweck 257
 Verwendungszweckaufbau 258
 Herkunft 259
 kundeneigener Funktionsbaustein 260
 Voraussetzung 172

W

Wertart 270
 Wertetabelle 58

Z

Zahllauf 237, 330
 Ausführungsdatum 238
 Buchung 238
 einplanen 241
 gesperrtes Konto 239
 Laufidentifikation 238
 Mindestbetrag 247
 Nachbearbeitung 238
 Regulierungsliste 240, 250
 Zahlung ausschließen 246
 Zahlungsträgererstellung 239, 253
 Zahlungsanordnung 333
 Zahlungsavis 298
 Zahlungsbedingung 85
 Zahlungsfreigabe 313

Zahlungsgruppierung 247
 Zahlungsprogramm 240, 243
 Zahlungsträger 237, 301
 Zahlungsträgerformat 253
 Zahlungsträgerprogramm
 generisches 253
 klassisches 239
 Zahlungsvorschlag 238
 Zahlungsvorschlagslauf 238
 Zahlweg 85, 246
 Zahlwegauswahl 244
 Zeile, steuerrelevante 142
 Zeitpunkt 171
 00 (*Sortierfeld füllen*) 255
 05 (*Referenzfelder füllen*) 255
 06 (*Referenzfelder füllen*) 255
 10 (*Formatparameter prüfen*) 255
 11 (*Formatparameter prüfen*) 255
 20 (*Start/File Header*) 256
 21 (*Start/File Header*) 256
 25 (*File Close/Open*) 256
 30 (*Auftrag*) 256
 31 (*Auftrag*) 256
 40 (*Ende*) 256
 41 (*Ende*) 256
 Zeitpunktbaustein 254
 Dateiname 256
 neue Datei 256, 264
 Sortierkriterium 255, 263
 Zusatzkontierung
 Enjoy-Transaktion 163
 kundeneigene 161
 Summentabelle 163
 ZX*-Include 24