



OpenOffice.org

Kassenbuch

Dokumentation und Administratoren-Handbuch

OpenOffice.org Kassenbuch
(OOoKB)

November 2006

Programmversion 1.0.2

M.I.C. Consulting

Dipl.Wirtsch.Ing Thomas Krumbein

Riederbergstrasse 92 • 65195 Wiesbaden

Tel.: 0611- 188 53 39 • Fax 0611 – 188 53 40

www.mic-consulting.de • tk@mic-consulting.de

© Thomas Krumbein, 2006, alle Rechte vorbehalten

Dokumentation und Administratorenhandbuch

OpenOffice.org Kassenbuch mit Projektverwaltung für nicht mehrwertsteuerpflichtige Organisationen.

Hinweis:

Diese Broschüre erläutert die Benutzung und die Installation des Programms, nicht die Wartung oder die Programm-Anpassung. Hierfür gibt es das Administratoren-Handbuch beziehungsweise eine ausführliche Programmdokumentation.

Lizenzhinweise:

Initialautor: Thomas Krumbein, e-mail: thomaskrumbein@openoffice.org

Copyright 2006 Thomas Krumbein, alle Rechte vorbehalten.

Das Programm wurde inspiriert von der E/Ü von Uwe Schlenther u. a. (http://www.schlenther.de/software/index.html?ooo_eur.html), der Code jedoch komplett neu geschrieben.

Das Programm entstand dank finanzieller Unterstützung der "**Jugendzentren Köln gGmbH**" in Köln (www.jugz.de).

Dieses Programm ist freie Software. Sie können es unter den Bedingungen der GNU General Public License, wie von der Free Software Foundation veröffentlicht, weitergeben und/oder modifizieren, entweder gemäß Version 2 der Lizenz oder (nach Ihrer Option) jeder späteren Version.

Die Veröffentlichung dieses Programms erfolgt in der Hoffnung, dass es Ihnen von Nutzen sein wird, aber OHNE IRGENDNEINE GARANTIE, sogar ohne die implizite Garantie der MARKTREIFE oder der VERWENDBARKEIT FÜR EINEN BESTIMMTEN ZWECK. Details finden Sie in der GNU General Public License.

Sie sollten eine Kopie der GNU General Public License zusammen mit diesem Programm erhalten haben. Falls nicht, schreiben Sie an die Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307, USA.

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version. This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307, USA.

Das Programm sowie der Makro-Code stehen unter der GPL (GNU Public License), einen kompletten Lizenztext erhalten Sie unter <http://www.fsf.org/licenses/gpl.html>, eine deutsche Übersetzung unter <http://www.gnu.de/gpl-ger.html>.

Die hier vorliegende Dokumentation steht unter Creative Commons Lizenz und darf vervielfältigt, verbreitet und öffentlich aufgeführt werden, wenn der Name des Autors genannt und der Inhalt nicht verändert wird (für Details siehe <http://creativecommons.org/licenses/by-nd/2.0/de/deed.de>)

Versionshistorie:

Version 0.9.1 vom 15. September 2006, Initialversion, © Thomas Krumbein, MIC-Consulting, Wiesbaden, e-mail: thomaskrumbein@openoffice.org

Version 1.0 vom 15. Oktober 2006, kleine Fehlerbereinigungen, Fertigstellung Dokumentation

Inhaltsverzeichnis

Hinweis:	2
Lizenzhinweise:.....	2
Versionshistorie:.....	2
1 Funktionsumfang / Aufgabenstellung.....	5
2 Installation und Inbetriebnahme.....	5
2.1 Makros zu den „vertrauenswürdigen Quellen“ hinzufügen.....	6
2.2 Lokalisierung von OpenOffice.org.....	7
2.3 Nummernkreise einrichten.....	8
2.4 Logos einbinden.....	10
2.5 Belegsperre aktivieren.....	11
2.5.1 Belegsperre setzen.....	11
3 Administrator-Aufgaben.....	12
3.1 Passwort.....	12
3.2 Sperrfelder zurücksetzen.....	13
3.3 Kontenverwaltung.....	13
3.3.1 Konten hinzufügen.....	14
3.3.2 Konto ändern.....	16
3.3.3 Konto löschen.....	17
3.3.4 Kontenbereiche neu schreiben.....	17
3.4 Projektverwaltung.....	18
3.4.1 Projekte löschen.....	18
3.5 Löschen von Datensätzen.....	18
3.5.1 Löschen einzelner Datensätze.....	18
3.5.2 Alle Datensätze löschen.....	19
4 Programm anpassen.....	20
4.1 Berichte individualisieren.....	20
4.1.1 lfd_Nummer drucken.....	22
4.2 Dialoge verändern.....	22
4.2.1 Projektdialog.....	22
5 Dokumentation.....	23
5.1 Programmfluss.....	24
5.1.1 Aufgabenstellung.....	24
5.1.2 Realisierung.....	25
5.2 Struktur der Datei.....	27
5.2.1 Start der Datei.....	29

5.2.2 Tabelle Stammdaten.....	29
5.2.3 Tabelle Formeln.....	33
5.2.4 Die Tabelle Konten:.....	35
5.2.5 Die Tabelle Projekte.....	36
5.2.6 Monatstabelle Januar bis Dezember.....	37
5.2.7 Tabelle lfd_Rechnungen.....	38
5.2.8 Tabelle Auswertungen.....	39
5.2.9 Tabelle Projektkosten.....	40
5.2.10 Tabelle Kontenauswertung.....	41
5.2.11 Tabelle Kassenabgleich.....	42
5.3 Struktur der Makros.....	43
5.3.1 Admin	44
5.3.2 Belege_bearb.....	44
5.3.3 Daten.....	45
5.3.4 Ergebnis.....	45
5.3.5 Helper.....	45
5.3.6 Init_Mod.....	46
5.3.7 Kasse	49
5.3.8 Konten.....	49
5.3.9 Main.....	50
5.3.10 Projekte.....	50
5.3.11 Rechnungen.....	51
5.4 Die Dialoge.....	51
5.4.1 Beleg.....	51
5.4.2 DlgListe.....	52
5.4.3 Initdlg.....	54
5.4.4 KtoDlg.....	55
5.4.5 ProjDlg.....	56
5.4.6 PWdlg.....	57
5.4.7 Statusbar und Wait.....	58
5.5 Die Makros.....	60

1 Funktionsumfang / Aufgabenstellung

Das OpenOffice.org Kassenbuch (OOoKB) dient nicht mehrwertsteuerpflichtigen Organisationen zur Erfassung und Überwachung der Ein- und Ausgaben in Form eines fortlaufenden Kassenbuches. In diesem werden alle Bar-Belege chronologisch gespeichert. Umfangreiche Berichtsfunktionen und Auswertungen lassen stets einen Blick für den aktuellen Kassenstand zu. Darüber hinaus können Rechnungen – also bargeldlose Transaktionen – erfasst und gesondert verbucht werden. Dies dient überwiegend der Vorbereitung und Sortierung einer nachfolgenden Buchhaltung. Mit Barbelegen und Rechnungen sind in der Regel alle Transaktionen einer kleineren Organisation erfasst und die Ergebnisse spiegeln die Geschäftstätigkeit komplett wieder. Neben unterschiedlichen Konten können auch noch Projekte separat erfasst und abgerechnet werden. Einzelne Projekte können auch wie Kostenstellen interpretiert werden – eine ausführliche Auswertung ist dann das Ergebnis.

Das ganze Programm basiert auf einer Calc-Tabelle in OpenOffice.org und ist komplett in OpenOffice.org Basic realisiert worden. Dabei wurde auf eine möglichst einfache Benutzerführung Wert gelegt, wobei diese fast ausschliesslich über Dialoge realisiert ist. Das bedeutet aber auch, dass direkte Eingaben in die Calc-Tabellen in der Regel nicht notwendig und nicht erwünscht sind. Einzige Ausnahmen: Der Kassenabgleich. Dort sind einige Zellen freigeschaltet und nehmen Benutzereingaben auf. Alle anderen Zellen und Tabellen sind für Eingaben gesperrt, dadurch wird eine mögliche Fehlbedienung ausgeschlossen.

Das Kassenbuch ist konsequent auf zwei Nutzergruppen hin programmiert:

Den „**normalen**“ **Bediener**, der Daten eingibt und Auswertungen vornimmt beziehungsweise druckt und den **Administrator**, der darüber hinaus wesentliche Einstell- und Konfigurationsarbeiten am Programm selbst vornimmt und dieses somit optimal zum jeweiligen Einsatz anpasst. Um in den Administrator-Modus umzuschalten, ist ein Passwort nötig, das der Administrator selbst setzen kann. Bei der Erstinstallation ist das Passwort auf „0000“ eingestellt – also viermal die Null.

Hinweis:

Im normalen Betrieb sollte das Kassenbuch nicht im Administrator-Modus betrieben werden, um versehentliche Änderungen innerhalb der Tabellen zu vermeiden.

2 Installation und Inbetriebnahme

Eine Installation des Programmes ist nicht erforderlich. Das OOoKB kommt als OpenOffice.org Calc Vorlagendatei und kann direkt aufgerufen werden.

Es macht jedoch Sinn, sich zuerst ein spezielles Verzeichnis zu erzeugen und dort dann die Vorlagendatei sowie alle anderen benötigten Dateien (siehe zum Beispiel „Logos“) unterzubringen – und dieses Verzeichnis auch zum Speichern der jeweiligen Kassenbücher zu verwenden.

Da das Kassenbuch Makro-gesteuert wird, macht es auch Sinn, den Pfad (also das gerade erzeugte Verzeichnis) in den Optionen von OpenOffice.org als „sicheres Verzeichnis“ aufzunehmen, dann entfällt der Bestätigungsdialog zum Ausführen der Makros.

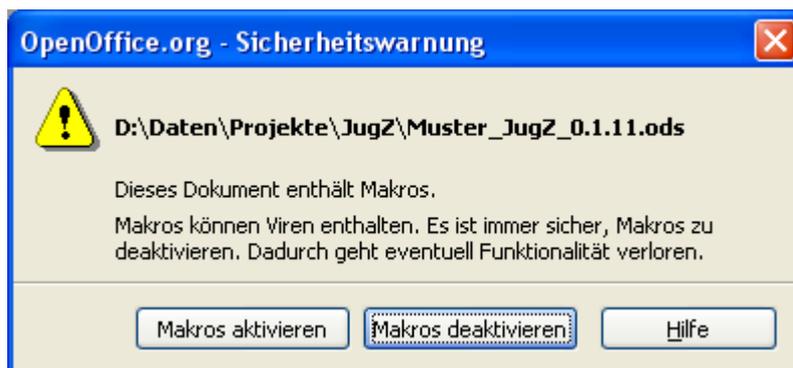
Weitere detaillierte Hinweise zum Programmstart und zu notwendigen Einstellungen finden sich in

der Bedienungsanleitung zum OpenOffice.org Kassenbuch und sind in der Regel über die Dialoge realisierbar.

Die jetzt hier gemachten Aussagen sollten entweder auf die Vorlage selbst oder vom Administrator für das erzeugte Kassenbuch vorgenommen werden.

2.1 Makros zu den „vertrauenswürdigen Quellen“ hinzufügen

Jedes mal beim Start von dem OOoKB wird der folgende Dialog erscheinen:



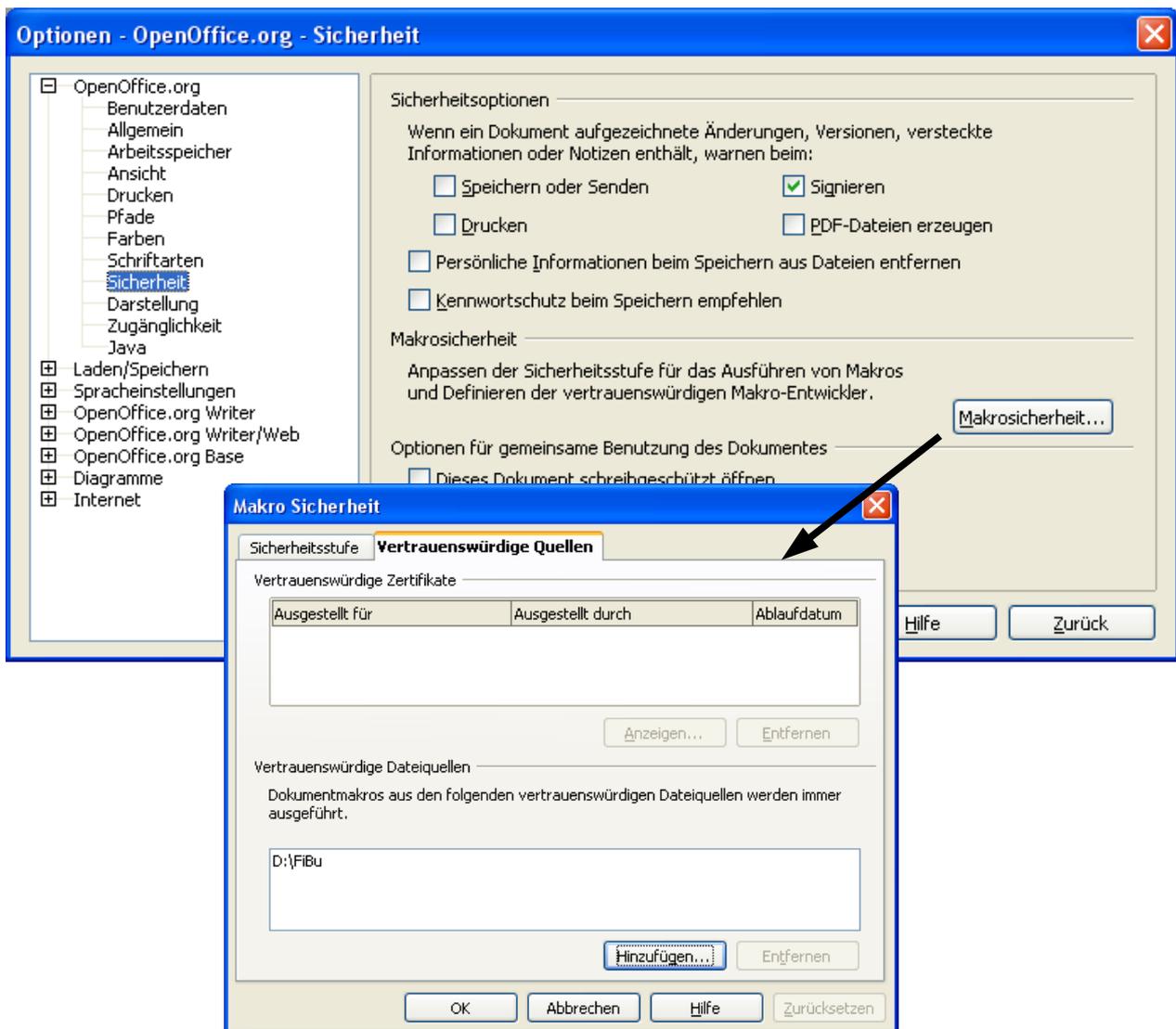
Hier müssen die Makros aktiviert werden, sonst ist das Kassenbuch funktionsuntüchtig. Es macht also durchaus Sinn, den Pfad zu den Kassenbuch-Dateien als „vertrauenswürdige Quelle“ zu fixieren, dann wird die Sicherheitsmeldung nicht mehr angezeigt und die Makros automatisch aktiviert.

Achtung:

Das Hinzufügen eines Pfades zu den „vertrauenswürdigen Quellen“ birgt allerdings auch Risiken! Da nur ein Verzeichnis als „vertrauwürdig“ eingestuft werden kann, können dort anschliessend natürlich auch „unseriöse“ Dateien gespeichert werden, bei denen dann ebenfalls ohne Nachfrage Makros aktiviert werden können. Makros können auch Code enthalten, der Schaden am System oder an Dateien erzeugen kann!

Um das Verzeichnis zu den vertrauenswürdigen Quellen hinzuzufügen, gehen Sie in die Optionen von OpenOffice.org (Menübefehl **Extras – Optionen**) und dort in den Bereich **OpenOffice.org – Sicherheit**.

Klicken Sie jetzt auf die Schaltfläche **Makrosicherheit**, wählen den Bereich „Vertrauenswürdige Quellen“, klicken auf die Schaltfläche **Hinzufügen** und wählen im sich öffnenden Pfadauswahldialog das Verzeichnis, in dem das OOoKB gespeichert ist. Nach der Bestätigung ist der Pfad jetzt in der Listbox sichtbar. Bestätigen Sie das Ganze mit **OK** und verlassen Sie die Optionen. Ab jetzt entfällt der Sicherheitsdialog beim Start des Kassenbuchs.

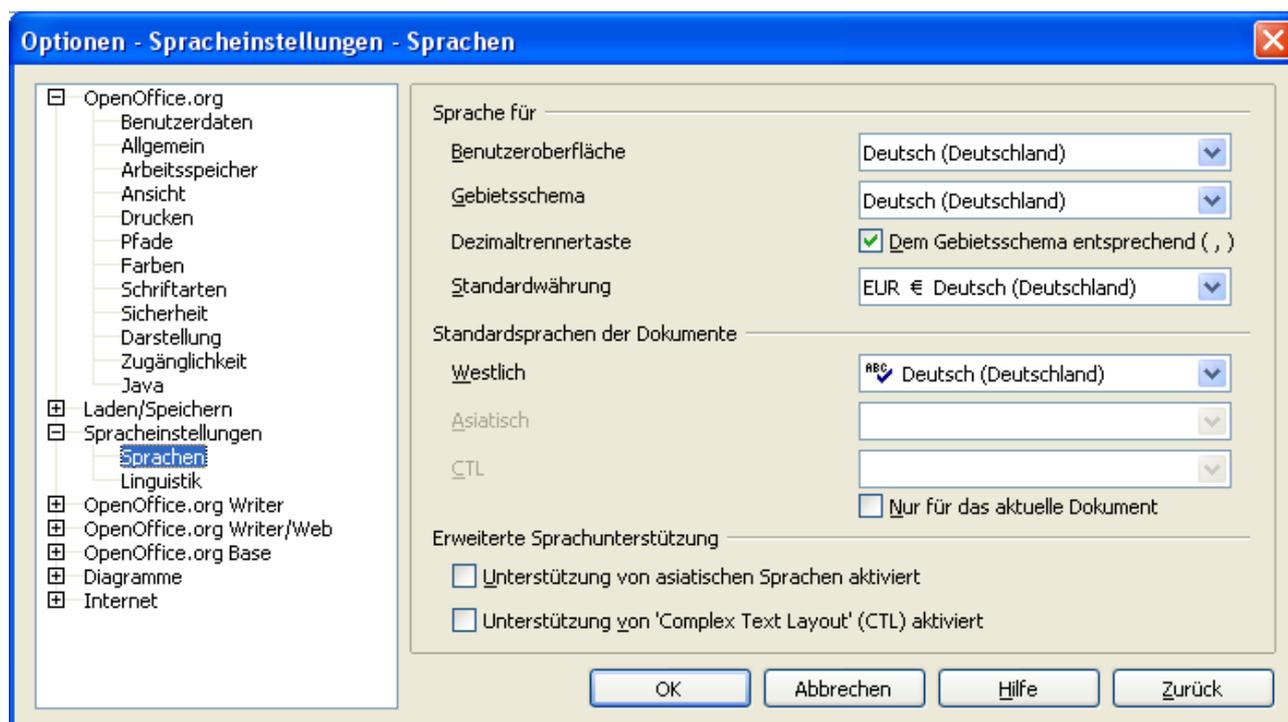


2.2 Lokalisierung von OpenOffice.org

Um den korrekten Einsatz des OpenOffice.org Kassenbuches zu gewährleisten, muss eine lokalisierte OpenOffice.org Version genutzt werden. Es reicht dabei nicht, dass die Oberfläche in „Deutsch“ erscheint beziehungsweise deutsche Menüs angeboten werden, auch der „innere“ Kern muss extra lokalisiert werden. Üblicherweise (im normalen Betrieb) ist dies nicht nötig, da OpenOffice.org dann die Einstellungen automatisch aus den Betriebssystem-Einstellungen entnimmt, leider tritt dieser Fall nicht bei Dialogen in Makros auf. Wurde das System nicht lokalisiert, so erscheinen die Währungsfelder alle mit einem „\$“-Zeichen statt dem gewünschten €-Zeichen.

Die Lokalisierung ist jedoch schnell durchgeführt:

Gehen Sie dazu in die Optionen (Extras – Optionen) und dort in den Bereich Spracheinstellungen – Sprachen. Bei der normalen Installation sind die Angaben dort alle auf „Standard“ gestellt. Ändern sie das nun manuell und explizit auf „Deutsch“ sowie die Standardwährung auf „EUR €“.



Jetzt zeigen auch die Währungsfelder der Dialoge das €-Zeichen an und nicht mehr das \$-Zeichen.

2.3 Nummernkreise einrichten

Das Kassenbuch verwendet unterschiedliche Nummernkreise für unterschiedliche Funktionen. Hier zunächst die Übersicht:

- ➔ **Ifd.Nr:** Jede Buchung erhält intern eine laufende Nummer, diese Nummer wird automatisch vom Programm vergeben und ist über Dialoge nicht veränderbar. In der ausgelieferten Vorlage beginnt diese Nummer bei 1, die Nummer kann vom Administrator jedoch geändert werden. Beachten Sie dabei jedoch folgendes:

Das Programm erhöht die Nummer automatisch um 1 bei jeder Buchung, Sie sollten also im laufenden Prozess diese nicht mehr verändern. Die Nummer muss eine Zahl (also ein Wert) sein, alpha-numerische Kombinationen sind nicht zulässig.

- ➔ **Beleg-Nummer:** Jeder gebuchte Beleg erhält eine fortlaufende Belegnummer. Die Belegnummer lässt sich auch über die Optionen in den Stammdaten einstellen – oder direkt in der Tabelle. Dabei kann die Belegnummer eine beliebige Alphanumerische Zeichenkette darstellen, der letzte vorhandene Ziffernblock wird intern dann immer hochgezählt. Die Belegnummer darf keine Leerzeichen enthalten! Beispiele für gültige Belegnummern: ABC2006DE001 (de 001 wird hochgezählt), 2006-001 (die 001 wird hochgezählt), ab001_meineNummer (die 001 wird hochgezählt), 1234001 – (die komplette Nummer wird hochgezählt)
- ➔ **Rechnungsnummern:** Auch alle Rechnungen erhalten eine eigene Rechnungsnummer, die ebenfalls intern hochgezählt werden kann. Auch die Rechnungsnummern lassen sich über den Stammdaten-Dialog einstellen. Für Rechnungsnummern können ebenfalls

beliebige Alphanumerische Zeichenfolgen genutzt werden, ohne Leerzeichen, und der letzte darin vorkommende Ziffernblock wird dann hochgezählt.

- ➔ **Projektnummern:** Die Projektnummer ist immer eine zwei- bis sechsstelligen Zahlenfolge, wobei führende Nullen erlaubt sind. Projektnummern können ebenfalls intern automatisch vergeben werden, die letzte verteilte wird hier angegeben. Die Projektnummernlänge wird intern über den Parameter `cAnzPrjNr` im Modul `Init` definiert und steht derzeit auf 6.
- ➔ **Kontonummern:** Für Kontonummern gibt es keine eigene Einstellmöglichkeit noch werden diese automatisch vergeben. Kontonummern sind ebenfalls immer vierstellig und rein numerisch, dies ist bei der Anlage neuer Konten zu beachten. Führende Nullen sind ebenfalls zulässig.

Alle Nummern (außer der Kontonummer) finden sich in der Tabelle **Stammdaten** und dort in den **Spalten K + L, Zeilen 2-14**. Sie lassen sich im Administratorenmodus dort direkt editieren und verändern.

Letzte laufende Nummer:	35
Letztes Eingabedatum:	15.09.2006
Letzte Belegnummer:	2006-027
Letzte Rechnungsnummer:	RE-2006-008
Letzte Projektnummer:	0004

Ebenfalls in diesem Block abgespeichert wird das Datum der letzten Eingabe (Rechnungs- oder Belegbuchung)

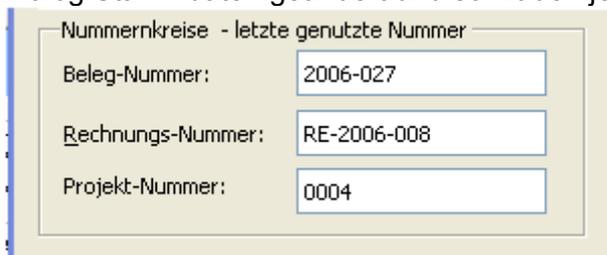
Dies dient zu Kontrollzwecken, wird aber auch als Vorgabedatum bei neuen Belege- oder Rechnungsbuchungen verwendet.

Werden keine eigenen Nummernkreise definiert, so verwendet das Programm intern fest voreingestellte Vorgabewerte, da die Nummernkreise nicht leer sein dürfen (siehe hierzu auch Modul `Init_Mod`).

Im Übrigen sollten Sie die Nummernkreise nur am Anfang fixieren und anschliessend nicht mehr ändern.

Es könnte sonst zu Inkonsistenzen der Daten führen.

Beleg-, Rechnungs- und Projektnummern können im Übrigen jederzeit von allen Benutzern im Dialog Stammdaten geändert und somit den jeweiligen Bedürfnissen angepasst werden.



Außerdem lassen sich natürlich auch die Nummern direkt beim Beleg- bzw. Rechnungserfassen manuell eingeben und somit die automatischen Nummern verändern oder überschreiben – das sollte aber ebenso nicht geschehen. Hier ist die Schulung der Mitarbeiter gefragt und eine entsprechende Kontrolle

notwendig.

Die in der Tabelle gespeicherten Nummern werden jeweils überschrieben, wenn entweder in den Stammdatendialog neue Nummern eingegeben werden oder bei der Beleg- oder Rechnungserfassung. Es wird immer die zuletzt benutzte Nummer gespeichert – neue Belege,

Rechnungen oder Projekte bekommen die gespeicherte Nummer + 1 zugewiesen.

2.4 Logos einbinden

Die Belegerfassungsmaske bietet die Möglichkeit, zwei Logos oder Bilder einzubinden. Hierzu sind zwei quadratische Flächen oben links und oben rechts vorgesehen.



Dabei ist das linke Feld größer (Kantenlänge 55 ma), das rechte Feld kleiner (Kantenlänge 44 ma). Die Einheit ma ist eine OoO spezifische, interne Einheit, die sich aus dem systemeigenen Standardfont berechnet. Es gibt somit keine direkte und einfache Umrechnung. Diese ist aber auch nicht nötig, da das Logo oder Bild in die Fläche eingepasst wird. Wichtig ist nur, dass es sich um ein quadratisches Bild oder Logo handelt (Kantenlängen müssen gleich sein), sonst wird das Bild verzerrt!

Erkannt und eingebunden werden alle Grafikformate, die OpenOffice.org so kennt, also zum Beispiel jpg, tif, bmp, gif, png, wmf, svg und viele mehr. Da die Hintergrundfarbe des Dialoges nicht weiss ist, sollten Sie bei nicht quadratischen Motiven auf Grafikformate mit Transparenz-Funktion zurückgreifen, also zum Beispiel gif oder png. Das ergibt dann ein optisch schöneres Bild.

Nochmal der Hinweis:

Logos oder Bilder sollten in einer quadratischen Bildform vorliegen, um Verzerrungen zu vermeiden!

Das Einbinden der Grafiken ist schnell erledigt: Die Grafikdateien werden einfach in das gleiche Verzeichnis kopiert, in der sich auch die verwendete OOoKB Calc-Datei befindet. Dann werden die Dateinamen in die Calctabelle Stammdaten in die Zellen L19 + L20 eingetragen – fertig.

Linkes Logo:	jugz.tif
Rechtes Logo:	katze.jpg

Achten Sie dabei auf Groß- und Kleinschreibung und – je nach Betriebssystem – auf die „richtige“ Dateibezeichnung.

Wichtig:

Die Logo- oder Grafikenamen müssen schon in URL Schreibweise vorliegen! Das bedeutet: Keine Leerzeichen, keine Sonderzeichen, lediglich Unterstrich erlaubt, sonst nur Buchstaben und Zahlen!

Möchten Sie die Logos nicht im gleichen Verzeichnis unterbringen, so ergänzen Sie den Dateinamen um den relativen Pfad ausgehend vom Verzeichnis des OOoKBs. Beispiel: Sie haben ein Unterverzeichnis Bilder angelegt, und dort sind Ihre Logos drin, so schreiben Sie in die Zellen:

/Bilder/jugz.tif und /Bilder/katze.ipg. auch hier gilt: die Schreibweise muss in URL-Manier erfolgen, also auch bei Windowssystemen kein Backslash als Verzeichnistrenner, sondern der Slash (Umschalt + 7).

Hinweis:

Die Logos können erst dann angezeigt werden, wenn die Programmdatei eine eigene URL Adresse besitzt, also wenn sie mindestens einmal gespeichert wurde. Haben Sie die Datei beim Start nicht gespeichert, so erscheint der Belegdialog ohne Logos!

2.5 Belegsperrung aktivieren

Das Programm besitzt eine Belegsperrung, die verhindert, dass Belege oder Rechnungen zurückliegender Buchungsperioden nachträglich geändert werden.

Dieses ist zum Beispiel dann sinnvoll, wenn man regelmäßig die (abgeschlossenen) Monatsberichte ausdruckt und archiviert, dann sollten die Beträge nachträglich nicht mehr geändert werden können. Denn durch geänderte Beträge ändern sich auch die Überträge aller folgenden Monate und somit auch die Journale.

Die Belegsperrung sorgt dafür, dass der normale Benutzer wesentliche Daten wie Konto, Betrag oder Datum eines Beleges nicht mehr ändern kann.

Dabei kann die Belegsperrung individuell festgelegt werden.

Um die **Belegsperrung zu aktivieren**, wird in die Zelle **L23** der Tabelle **Stammdaten** eine „1“ eingetragen (Voreinstellung) – um die Belegsperrung zu deaktivieren eine „0“ (Null).

Zurückliegende Monate sperren:	1	1 = ja, 0 = Nein
Monate zurück:	2	1 – 11
letzter gesperrter Monat:	1	1 – 12

Ist die Belegsperrung aktiviert, kann man auch noch festlegen, welche **Monate** ausgehend vom aktuellen **rückwirkend** gesperrt werden. Diese Festlegung geschieht in Zelle **L24** der Tabellen Stammdaten, und hier lassen sich Zahlen von 1 bis 11 eintragen, wobei „1“ bedeutet, dass nur der aktuelle Monat nicht gesperrt ist, alle zurückliegenden aber schon. Tragen sie eine „2“ ein, so wird der aktuelle und der direkt davorliegende Monat nicht gesperrt, allen anderen werden gesperrt. Bei „3“ wird der aktuelle sowie die beiden davorliegenden nicht gesperrt und so weiter.

Die letzte Zeile der Belegsperrung gibt Auskunft darüber, welcher Monat der derzeit letzte gesperrte Monat ist. Diese Zeile kann zwar manuell geändert werden – zum Beispiel für Wartungsarbeiten – wird aber immer wieder vom Programm überschrieben, wenn die Belegsperrung gesetzt wird. Und der Wert ist dann lediglich abhängig von den beiden anderen Werten sowie des auslösenden Ereignisses.

2.5.1 Belegsperrung setzen

Die Belegsperrung zurückliegender Monate wird gesetzt beim erstmaligen Ausdruck eines Monatsjournals. Dabei wird der aktuell ausgedruckte Monat als Basis genutzt und von diesem Wert (Monatszähl) die „Monate zurück“ abgezogen. Dieser Wert dann wiederum wird zurückgeschrieben in die Tabelle Stammdaten auf die Position **L25**, deren Wert wiederum ausgewertet wird und bestimmt, ob ein Beleg gesperrt ist oder nicht.

Beim erstmalige Drucken einer Monatstabelle wird in die Zelle **I1** der entsprechenden Monatstabelle eine „1“ eingetragen – diese Kennzahl sagt aus, dass bereits erstmalig gedruckt wurde. Steht die „1“ bereits dort, wird der Sperrmechanismus nicht mehr durchgeführt.

Siehe hierzu aber auch Modul **Helper - Journal_drucken**.

3 Administrator-Aufgaben

Ist das Programm erst einmal eingerichtet und den eigenen Wünschen entsprechend angepasst, können in der Regel alle anderen Arbeiten im Benutzermodus erfolgen. Das Verbuchen aller Vorgänge, das Anlegen neuer Projekte, das vergeben von Budgets, die Auswertungen und das Drucken derselben lässt sich alles über die Benutzerschnittstelle erledigen. Auch das Ändern falsch eingegebener Daten ist möglich (solange diese noch nicht gesperrt sind), ein Administrator hat also in der Regel keine weiteren Aufgaben. Alle Details zu möglichen Benutzeraktionen sind ausführlich im Bedienungshandbuch beschrieben.

Immer nur dann, wenn tiefgreifende Änderungen durchzuführen sind, ist der Administrator gefragt. Der Administrator kann alle Tabellen anpassen, Änderungen im Programmcode vornehmen und vieles mehr.

3.1 Passwort

Um das getrennte Arbeiten von Benutzer und Administrator auch auf verschiedene Personen übertragen zu können, besitzt das Programm eine Passwortsperrung für den Administratormodus. Ohne das Passwort kann man den Admin-Modus nicht aktivieren und den Tabellenschutz somit aufheben.

Das Passwort selbst ist dabei an zwei verschiedenen Orten untergebracht, einmal OOO programmintern (darauf besteht nur wenig Einfluss und hier sollten Sie auf keinen Fall manuell eingreifen) und einmal in der Tabelle **Formeln** im Programm selbst. Nur hier dürfen sie das Passwort ändern oder löschen, auf diese Zelle greifen die Makros zu und steuern die Zugänglichkeit.

Um die Tabelle Formeln überhaupt sichtbar zu schalten, müssen Sie zunächst den Administratormodus einschalten. Das vorgegebene Passwort lautet:

Admin-Passwort: 0000 (vier mal die Null)

Durch das Umschalten in den Admin-Modus werden alle Tabellensperrungen (Schutz) aufgehoben und die Tabelle **Formeln** wird eingeblendet. In dieser befindet sich neben dem Passwort auch noch ein Verwaltungsmenü für Administratoren.

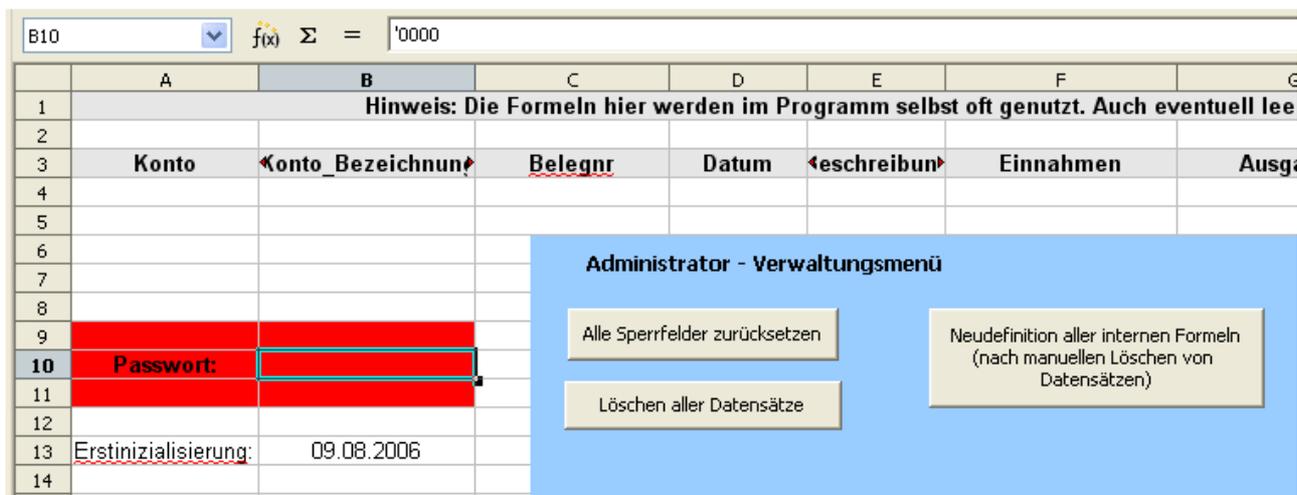


Das Passwort selbst steht in Zelle **B10**, ist jedoch in der gleichen Schriftfarbe formatiert wie der Zellhintergrund, insofern auch nicht direkt lesbar. Das verhindert, dass das Passwort zufällig „durch Blick über die Schulter“ ausgelesen werden kann. Markiert man die Zelle, ist das Passwort in der Eingabezeile sichtbar und kann natürlich geändert werden.

Wichtig:

Das Passwort muss als Text in der Zelle stehen – bei Verwendung von nur Zahlen (Ziffern) müssen Sie unbedingt zuerst das einfache Hochkomma eingeben ('), um die Eingabe als Text zu kennzeichnen.

Nutzen Sie alphanumerische Zeichenfolgen ohne Sonderzeichen und ohne Leerzeichen – und beginnen Sie ihr Passwort mit einem Buchstaben – dann gibt es keine Probleme. Sie können auf tiefe Erkenntnisse der Kryptologie verzichten – der Passwortschutz ist nicht extrem sicher und „Hacker“ können ihn aushebeln. Er dient lediglich zur Trennung von Benutzer und Administrator.



The screenshot shows a spreadsheet with the following data:

	A	B	C	D	E	F	G
1	Hinweis: Die Formeln hier werden im Programm selbst oft genutzt. Auch eventuell leer						
2							
3	Konto	Konto_Bezeichnung	Belegnr	Datum	Beschreibung	Einnahmen	Ausgaben
4							
5							
6							
7							
8							
9							
10	Passwort:						
11							
12							
13	Erstinizialisierung:	09.08.2006					
14							

The Administrator - Verwaltungsmenü overlay contains the following buttons:

- Alle Sperrfelder zurücksetzen
- Neudefinition aller internen Formeln (nach manuellen Löschen von Datensätzen)
- Löschen aller Datensätze

In Zelle B13 wird das Datum der Erstinizialisierung gespeichert – nur zu Informationszwecken. Das ist das Datum, an dem Sie die Vorlage aufrufen und eine *.ods Datei erzeugen.

3.2 Sperrfelder zurücksetzen

Im Administratoren Menü gibt es aktuell drei Schaltflächen – eine davon setzt die Sperrfelder zurück. Die beiden anderen werden später noch besprochen.

Sie erinnern sich: Der Administrator kann in der Tabelle Stammdaten eine Belegsperrung aktivieren sowie die zurückliegenden Sperrmonate bestimmen.

Ist die Belegsperrung aktiviert, erhalten die Monatstabellen im Feld **I1** einen Flag, ob beim Drucken die Sperre aktiviert werden soll oder nicht. Gleichzeitig wird in der Tabelle **Stammdaten** der Monat eingetragen, der als letzter gesperrt wurde (in Feld **L25**). Über den Button **Sperrfelder zurücksetzen** werden genau diese Felder zurückgesetzt, es wird jeweils eine 0 eingetragen.

Das bedeutet, kein Monat ist mehr gesperrt, alle Sperrflags der Druckfunktion sind zurückgesetzt. Die Sperre wird wieder aktiviert, wenn das nächste mal ein Monatsjournal gedruckt wird.

3.3 Kontenverwaltung

Neue Konten erzeugen oder bestehende Konten in Ihren wesentlichen Daten ändern kann nur der Administrator. Diese Einschränkung erscheint notwendig, um dem Benutzer hier nicht komplett freie Hand zu lassen.

Dennoch sollte auch der Administrator die Änderungen nur über die Dialogmasken vornehmen um Fehler zu vermeiden.

3.3.1 Konten hinzufügen



	B	C	
1			
2			
3		Neues Konto	
4			
5			
6	Art	Kontonummer	
7	E	1658	dlfd.Schecks

Um ein neues Konto hinzuzufügen klickt man auf den entsprechenden Button entweder in der Tabelle **Stammdaten** (links) oder in der Tabelle **Konten** (rechts). In beiden Fällen erscheint zunächst der Passwort-Dialog, falls man nicht schon als Administrator angemeldet ist. Nach Eingabe des richtigen Passwortes wird der entsprechende Modus eingeschaltet und es öffnet sich die Konto-Eingabe Maske.

Hier kann jetzt ein neues Konto hinzugefügt werden.



Die Informationen sind selbsterklärend.

Jedes Konto kann nur eine **Art** darstellen: entweder ein Einnahme- oder ein Ausgabekonto. Die Wahl hier bestimmt die Darstellung in der Belegmaske.

Als **Kontonummer** muss eine maximal vierstellige, numerische Eingabe erfolgen. Führende Nullen sind zulässig.

Die **Kontobezeichnung** ist frei wählbar als Text.

Das **Budget** ist optional und kann auch später durch den Benutzer gesetzt oder verändert werden.

Des weiteren kann ein Konto **inaktiv** geschaltet werden – dann ist es im normalen Betrieb nicht mehr auswählbar, es können keine Buchungen mehr darauf erfolgen. Die Standardeinstellung ist „aktiv“, man markiert ein Konto beispielsweise als inaktiv, wenn schon Buchungen über dieses Konto gelaufen sind, zukünftig dieses aber nicht mehr genutzt werden soll. Würde das Konto jetzt

gelöscht, wäre die Buchhaltung inkonsistent.

Mit der zweiten Option kann man entscheiden, ob das Konto Teil der zusammenfassenden Auswertung ist oder nicht. Die Option kann jederzeit geändert werden.

Mit **Daten speichern** wird das Konto erzeugt und in die Kontenliste eingetragen – und zwar am Ende!

Art	Kontonummer	Kontobeschreibung	Aktiv	Auswertung	Budget	Summen	In_Prozent	Restbudget
E	1658	<u>dlfd Schecks</u> (Einnahmen)	1	0	125.000,00 €	0,00 €	0,00%	125.000,00 €
E	5500	Eigeneinnahme Projekte	1	1		690,00 €		
E	5701	Erlöse Raumvermietung	1	1		0,00 €		
E	5510	Erlöse Veranstaltungen	1	1		0,00 €		
E	5530	Erlöse aus Verzehr	1	1		182,00 €		
E	5750	Erlöse Telefon	1	1		45,00 €		
E	5705	Erlöse aus Vermietung <u>stfr</u>	1	1		0,00 €		
E	5505	Elternbeiträge	1	1		0,00 €		
E	5550	Erlöse <u>Essensgelder</u> ÜMB	1	1		0,00 €		
E	5515	Kegelbahn (Einnahmen)	1	1		0,00 €		
E	5610	Spenden	1	1	1.000,00 €	111,00 €	11,10%	889,00 €
E	5733	Erlöse Fortbildung/Seminare	1	1		0,00 €		
E	5755	sonstige <u>Erstattungen</u>	1	1		0,00 €		
E	0300	leer 3	1	0		0,00 €		
A	1658	Einzahlung Bank/Kasse (Ausgaben)	1	0		0,00 €		
A	0721	Einrichtungsgegenstände ab 475,60 € Bru	1	1		50,00 €		
A	0722	Hausausstattung (Möbel, Geschirr,...) ab 479	1	1		0,00 €		

Der Administrator kann auch direkt in der Liste Konten ändern oder neue Konten hinzufügen.

Wichtig sind folgende Punkte:

Die Spalte B bestimmt die Art des Kontos. Zulässig sind nur die Eingaben „E“ für Einnahmekonto oder „A“ für Ausgabekonto. Je nach Eingabe färbt sich der Zellhintergrund grün (für Einnahme) oder rot (für Ausgabe).

Die Spalte C nimmt die Kontonummer auf – wie gesagt, ein numerischer Wert, maximal 4-Stellig, also im Bereich 1 – 9998 (das Konto 9999 ist schon intern andersweitig belegt). Angezeigt wird die Kontonummer 4-stellig mit führenden Nullen.

In die Spalten E (Aktiv) und F (Auswertung) werden nur die Werte 1 (für eingeschaltet) und 0 (ausgeschaltet) eingetragen. Ein Budget ist optional.

Die Spalten G (I) wird später vom Programm aufgefüllt, hier sollte man nichts eintragen.

Die Spalten J und K beinhalten Formeln, welche automatisch geschrieben werden, wenn man ein Konto über die Maske erzeugt. Beim manuellen Eintrag müssen sie auch manuell eingetragen werden, falls gewünscht. Am einfachsten kopiert man die Formeln natürlich aus einer vorhandenen Kontozeile, man kann sie aber auch direkt eingeben:

Neues Konto in Zeile 57:

Spalte J (In_Prozent): =WENN(H57; I57/H57;"")

Spalte K (Restbudget) =WENN(H57;H57-I57;"")

Wichtig:

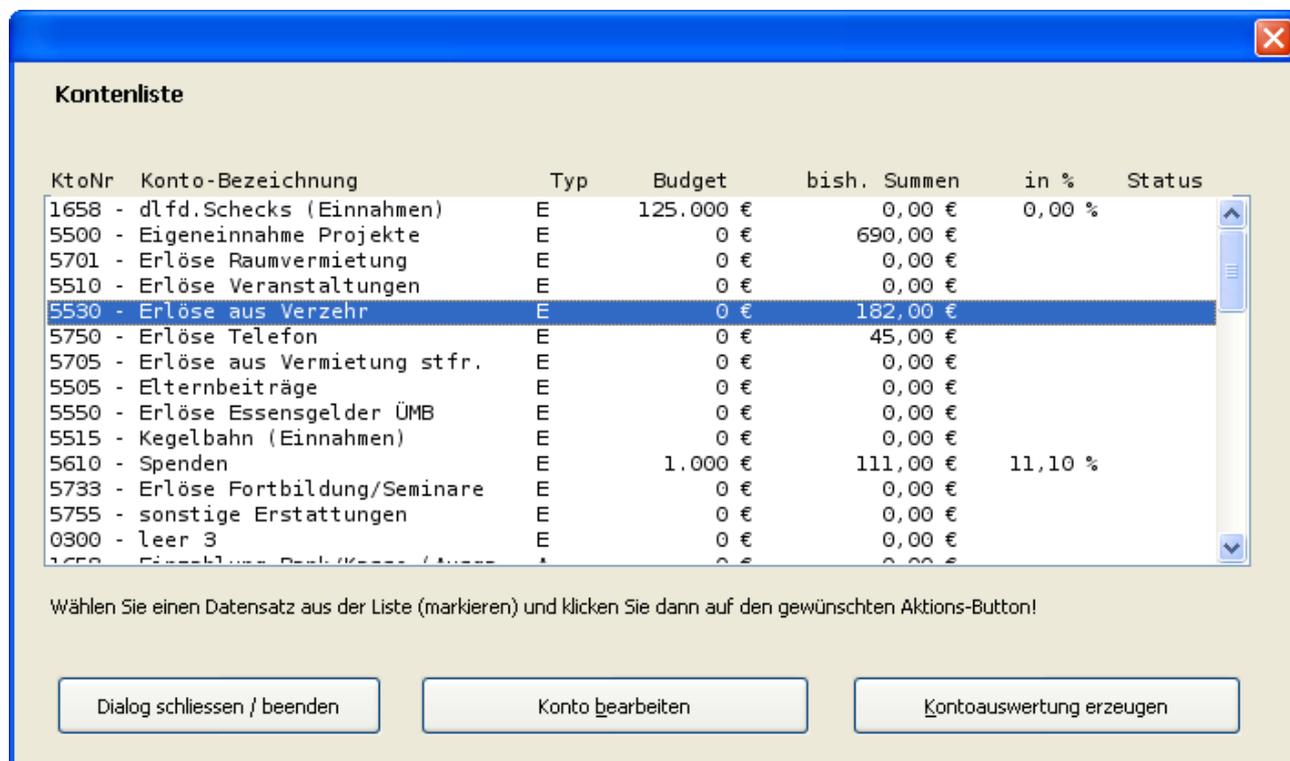
Neue Konten müssen nahtlos an die vorhandenen angefügt werden. Es darf keine Leerzeile entstehen. Das Programm wertet die Spalten B und C (Kontonummer) später aus und stoppt bei der ersten leeren Zelle!

Wichtig:

Vergessen Sie nicht die Kontenbereiche neu zu schreiben nach dem Hinzufügen von Konten! Egal, ob Sie diese manuell oder per Dialog hinzugefügt haben..

3.3.2 Konto ändern

Ein bestehendes Konto kann der Administrator ebenfalls sehr einfach per Dialog ändern. Dazu wird zunächst der Konto-Auswahldialog gestartet (über die entsprechenden Buttons – Konto verwalten), hier wählt man nun das gewünschte Konto (markieren) und klickt dann auf **Konto bearbeiten**.



Es öffnet sich jetzt der Konto bearbeiten Dialog, die Kontendaten sind eingetragen und können vom Administrator beliebig geändert werden.



In diesem Fall ist ein Neuschreiben der Kontenbereiche nicht notwendig.

3.3.3 Konto löschen

Möchte man ein Konto löschen, so geht dies nicht über Dialoge. Hierzu markiert man in der Tabelle **Konten** die Zeile des Kontos und entfernt diese komplett über den Menübefehl **Bearbeiten – Zellen löschen**.

Wichtig:

Gehen Sie immer so vor! Löschen Sie immer die komplette Zeile. Nur dadurch wird gewährleistet, dass die Kontenliste zusammen bleibt und keine Leerzellen entstehen.

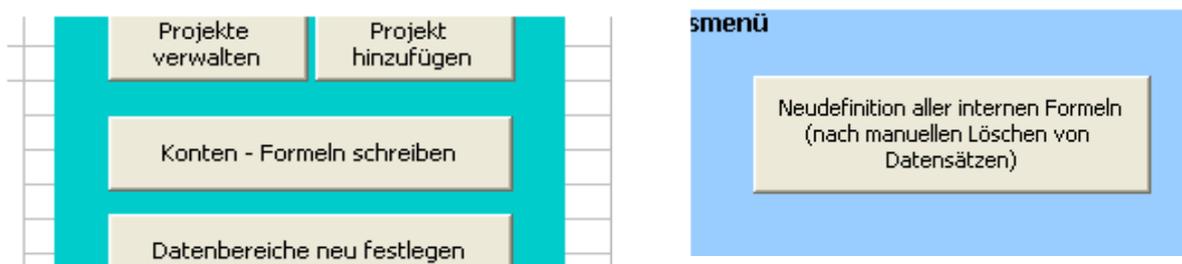
Nach dem Löschen einzelner Konten müssen unbedingt die Kontenbereiche neu geschrieben werden (siehe 3.3.4) sonst ist das Programm nicht mehr in der Lage, die Daten richtig zu verbuchen.

Hinweis:

Es ist selten eine gute Idee, ein Konto im laufenden Betrieb zu löschen. Wurden bereits Buchungen darauf vorgenommen, wären jetzt Inkonsistenzen im Programm. Dann ist es besser, das Konto bestehen zu lassen und es nur inaktiv zu schalten.

3.3.4 Kontenbereiche neu schreiben

Nach dem Hinzufügen oder dem Löschen von Konten müssen interne Formeln neu geschrieben beziehungsweise angepasst werden. Da die manuelle Anpassung sehr aufwendig ist, gibt es auch hierfür ein passendes Makro. Dieses muss unbedingt aufgerufen werden, bevor man mit dem Programm weiterarbeitet.



Hierzu klickt man auf den entsprechenden Button im Administrationsmenü in der Tabelle Formeln (rechts) oder ersatzweise auf den entsprechenden Button im Verwaltungsmenü der Tabelle Stammdaten (links).

Dadurch werden die jetzt vorhandenen Konten einzeln eingelesen, entsprechende Bereichsadressen in die Tabelle Formeln geschrieben und die Formeln der Auswertung angepasst. Details siehe hierzu im Abschnitt Dokumentation - Konten.

3.4 Projektverwaltung

Ähnlich wie die Konten werden auch die Projekte verwaltet. „Projekte“ steht dabei für einen Überbegriff, es könnten sich auch Kostenstellen oder ähnliches darunter verbergen.

Projekte stellen somit neben den Konten eine zweite Kategorie zur Buchung von Vorfällen dar. So ließe sich beispielsweise eine Barausgabe sowohl auf dem Konto „Fahrtkosten“ buchen, als auch dem Projekt „Mein erstes Projekt“ zuweisen. Beide Kategorien können Budgets erhalten und können getrennt voneinander ausgewertet werden.

Hinweis:

Aktuell wird die Projektverwaltung nur in Verbindung mit bestimmten Konten aktiviert. Die Liste der Konten ist „hart“ programmiert. Aktuell wird sie nur bei den Konten **6620** (Ausgabekonto) und **5500** (Einnahmekonto) aktiviert. Die Liste kann manuell erweitert werden, siehe hierzu Dokumentation – **Modul Helfer**, Routine **checkProjekt**.

Das Hinzufügen und verändern von Projekten erfolgt normalerweise direkt vom Benutzer aus, mit Hilfe der entsprechenden Dialoge. Hier gibt es keine Administratorspezifischen Aufgaben oder Besonderheiten.

Einzelheiten zum Erzeugen und Ändern von Projekten finden sich in der Bedienungsanleitung des Kassenbuches.

3.4.1 Projekte löschen

Um ein Projekt komplett zu löschen, gehen Sie genauso vor wie bei den Konten: Löschen Sie immer die komplette Zeile. Auch in der Projektliste darf es keine Leerzellen oder -zeilen geben, die darunter liegenden Projekte würden nicht mehr vom Programm erkannt werden.

Ausgewertet wird die Spalte A, Projektnummer.

Wichtig:

Auch hier gilt: Ein einmal verwendetes Projekt darf nicht mehr gelöscht werden – es kommt sonst zu Inkonsistenzen im Programm. Besser ist es, das Projekt inaktiv zu schalten.

Projekte können gefahrlos gelöscht werden nach dem Löschen aller Datensätze – dann ist kein Projekt mehr benutzt.

3.5 Löschen von Datensätzen

Für jede Buchung wird ein Datensatz erzeugt – und zwar im Monatsjournal des Buchungsdatums. Die Buchung wird immer als neue Buchung unten an die Liste angefügt, auch wenn das Buchungsdatum rückwirkend datiert wird. Die Journalliste entspricht also nicht dem Datumsverlauf, sondern dem Buchungsverlauf. Jede Buchung erhält zusätzlich eine laufende Nummer, die auch nicht mehr geändert werden kann.

3.5.1 Löschen einzelner Datensätze

Den Grundzügen der ordnungsgemäßen Buchführung nach ist es nicht zulässig, eine einmal durchgeführte Buchung zu löschen. Entsprechend diesen Grundsätzen ist lediglich eine Stornierung des Betrages zulässig.

Wenn der Administrator dennoch eine Buchung löschen möchte, so sind folgende Hinweise zu

beachten:

Im Monatsjournal ist immer die komplette Zeile zu löschen. Zwischen den einzelnen Journalzeilen darf keine Lücke entstehen – keine Leerzellen, keine Leerzeilen.

Ausnahme: Ist nur noch ein Datensatz im Journal übrig, darf die Zeile nicht entfernt werden. Zwischen Kopfzeile und Summenzeile muss immer eine Zeile übrigbleiben. Möchte man auch den letzten Datensatz löschen, so dürfen nur die Zelleninhalte entfernt werden. Da aber auch hier gilt, eine Leerzeile ist nicht zulässig, sollte zum Beispiel in das Feld „Belegnummer“ etwas eingetragen werden (vom Programm her wird der Begriff „Dummy“ eingetragen, es ist aber eigentlich egal, was Sie hineinschreiben).

Wichtig: Schreiben Sie nichts in die Spalte **A**, die Kontonummer. Diese **muss** leerbleiben!

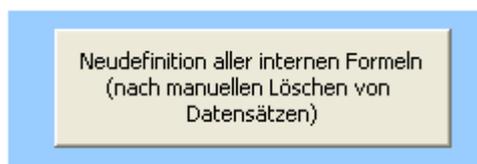
Durch das Löschen eines Datensatzes sind die Werte der Auswertungen nun aber inkonsistent (es sei denn, die Beträge waren jeweils Null).

Da jeder Datensatz beim Buchen nicht nur in das Monatsjournal eingetragen wurde, sondern auch mit seinem Betrag in der Konto- und Projektsalden Summierung, stimmen diese Werte jetzt nicht mehr.

Weiterhin wird für jeden Monat intern ein benannter Bereich geführt, der den Umfang des Journalzeilen abbildet und Grundlage für die Auswertungen darstellt (siehe auch Dokumentation – Programmfluss). Dieser benannte Bereich muss nun entweder manuell oder per Makro angepasst werden, um die Programmintegrität wieder herzustellen.

Vergessen Sie also nicht, nach dem Löschen einzelner Datensätze die Funktion **Datenbereiche neu schreiben** aufzurufen sowie die jeweiligen Funktionen **Projektsalden aktualisieren** und **Kontensalden aktualisieren**.

Alle diese Funktionen werden nacheinander aufgerufen über den Button im Verwaltungsmenü der Tabelle **Formeln**:



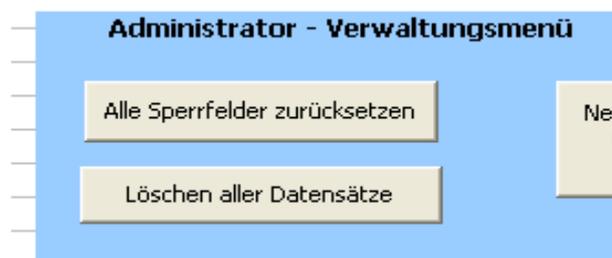
Wichtig:

Nach dem manuellen Löschen von Datensätzen müssen die internen Formeln neu definiert werden, sonst funktioniert das Programm nicht mehr!

3.5.2 Alle Datensätze löschen

Auch wenn es oft einfacher ist, ein neues Jahr aus der Vorlage zu erzeugen, kann es manchmal der Fall sein, dass man die alten Daten (Konto, Projekte, Stammdaten) behalten möchte. In diesem Fall können alle Bewegungsdaten (also Buchungen und Rechnungen) in einem Zug gelöscht werden.

Hierzu nutzt man den Button Löschen aller Datensätze im Verwaltungsmenü der Tabelle Formeln.



Alle Datensätze werden jetzt unwiderruflich gelöscht, die Salden der Konten und Projekte zurückgesetzt, die internen Formeln neu geschrieben, Sperrfelder werden zurückgesetzt ebenso die laufende Nummer und das letzte Eingabedatum.

Nicht geändert werden aber die Rechnungs- und Belegnummern, diese werden weitergeführt. Hier ist ein manueller Eingriff nötig – falls gewünscht.

Achtung:

Das Löschen der Datensätze ist unwiderruflich. Sie sollten unbedingt vorher eine Sicherungskopie Ihrer Daten durchführen (speichern unter einem anderen Namen), falls sie einen Jahreswechsel auf diese Art planen.

Damit sind die typischen Administratorarbeiten zur Genüge beschrieben worden. Alle weiteren Eingriffe und Anpassungen sind technisch auf der Makroebene oder bei den Einstellungen zu erledigen und bedürfen detaillierter Kenntnisse des Programms und der Programmierung.

4 Programm anpassen

Auch wenn das Kassenbuch schon für viele Anwendungen so passen dürfte, kann es doch die ein oder andere Einstellung geben, die man gerne anders hätte. Da das Programm unter der GPL steht, kann es auch von jedem selbst verändert werden. Ich unterscheide hierbei von einfachen „Anpassungen“ des bestehenden Programms (optischer Natur) und den „tiefen“ Änderungen – im Programmcode.

Für die „einfachen“ Anpassungen gebe ich hier Tipps.

4.1 Berichte individualisieren

Das Kassenbuch kennt verschiedene Möglichkeiten, Auswertungen zu drucken. Allen Ausdrucken gemeinsam ist die Notwendigkeit, dass in OpenOffice.org ein Drucker betriebsfertig eingebunden ist.

Grundsätzlich ist kein Druckbereich im Dokument definiert – wird also das Dokument über die „normale“ Druckfunktion von OpenOffice.org ausgedruckt, so werden über 30 Seiten gedruckt – teilweise gestückelt und nicht sinnvoll verknüpft. Man sollte dieses vermeiden.

Die Monatsjournale, das Rechnungsjournal, die Auswertungen sowie der Kassenbuchabgleich haben eigene Druckfunktionen als Buttons am Kopf der Tabelle definiert. Diese sollte man auch nutzen.

Dadurch wird per Makro ein Druckbereich definiert, die Kopf- und Fußzeilen entsprechend angepasst und der Bericht dann auf die Breite eines DIN A4 Blattes angepasst. Reicht das Blatt nicht, werden Folgeblätter gedruckt, wobei die Kopfzeile wiederholt wird.

Den jeweiligen Druckseiten liegen Seitenvorlagen zugrunde, welche die globalen Einstellungen speichern. Dabei stehen folgende Seitenvorlagen zur Verfügung:

Seitenvorlage	Besonderheiten	benutzt von:
Standard	keine Besonderheiten, keine Formatierung. Kopf- und Fusszeile eingeschaltet, Datum und Seitenzahl in Fußzeile.	auf allen nicht spezifizierten Tabellen – derzeit: Stammdaten, Projekte, Konten
Bericht	Querformat, Kopf- und Fusszeile, Kopfzeile links wird per Makro gesetzt (Einrichtungsname), rechts Datum und Uhrzeit, Fusszeile eingeschaltet, Seitenzahl, Bericht wird auf eine Seite (Breite) eingepasst, Ränder: Oben 1,5 cm, links/rechts 1,2 cm, unten 1,5 cm. Wiederholzeilen per Makro gesetzt	allen Monatstabellen (Januar – Dezember), lfd_Rechnungen, Kassenabgleich
Auswertung	Querformat, Kopfzeile, Kopfzeile links wird per Makro gesetzt (Einrichtungsname), rechts Datum und Uhrzeit, Fusszeile nicht eingeschaltet, Bericht wird auf eine Seite (Breite) eingepasst, Ränder: Oben 1,8 cm, links/rechts 1,1 cm, unten 1,5 cm. Wiederholzeilen per Makro gesetzt	Auswertungen
Projektauswertung	DIN A4 Hochformat, Kopf- und Fusszeile gesetzt, Kopfzeile links wird per Makro gesetzt (Einrichtungsname), rechts Datum und Uhrzeit, Fusszeile eingeschaltet, Seitenzahl, Bericht wird auf eine Seite (Breite) eingepasst, Ränder: Oben/unten 1,5 cm, links 2 cm, rechts 1,5 cm. Wiederholzeilen per Makro gesetzt	Projektkosten Kontoauswertung

Diese Vorlagen können natürlich individuell angepasst werden. Beachten Sie jedoch, dass teilweise die Kopf- und Fusszeilen per Makro gesetzt werden. Eingaben in den Vorlagen werden dann ohne Rückfrage überschrieben.

Tipp:

Alle Berichte erhalten in der Kopfzeile eine Kombination aus den Inhalten der folgenden Zellen aus der Tabelle Stammdaten: C7 (Name der Einrichtung) und C8 (Kostenstelle oder Details). Es ist also ohne weiteres möglich, dort die gewünschten Inhalte unterzubringen.

4.1.1 lfd_Nummer drucken

Jede Buchung (Beleg oder Rechnung) erhält intern eine fortlaufende Nummer und ist somit eindeutig identifizierbar. Jede Nummer wird nur einmal vergeben. Eine in sich konsistente Buchhaltung sollte alle Nummern fortlaufend ohne Lücken aufweisen.

Diese fortlaufende Nummer ist die letzte Spalte in den Monatsjournalen und in der Rechnungsübersicht. Sie kann mit ausgedruckt werden (empfohlen) oder auch nicht. Die Einstellung hierfür ist jedoch nur direkt im Code möglich.

Hierzu ruft man das Modul **Init_Mod** auf und stellt – ziemlich am Anfang – den entsprechenden Schalter wunschgemäß ein (Vorgabe: lfd_Nummer wird mit ausgedruckt):

```
'globale Konstanten definieren
REM Monatslisten
Global Const cMonStartRow = 3 'Erste Datenzeile der Monatslisten (Kopfzeile) -Achtung: Index!
Global Const cMonEndBlockRows = 5 'Zeilenanzahl des Schlussblocks
Global Const cMonStartColumn = 0 'Erste Spalte des Datenbereiches (Index)
Global Const cMonEndColumn = 8 'Letzte Spalte des Datenbereiches (Index)
Global Const cSaldoVorAdrE = "$F$1" 'Zelladresse des Saldovortrages Einnahmen
Global Const cSaldoVorAdrA = "$G$1" 'Zelladresse des Saldovortrages Ausgaben
Global Const cDruckLfdNr = 0 'Laufende Nummer wird gedruckt (1) oder nicht (0)
```

Zuständig für den Ausdruck der Spalte „Laufende Nummer“ ist die Konstante **cDruckLfdNr**, zu finden als letzten Eintrag im Block „Monatslisten“.

Diese Konstante kann zwei Werte aufnehmen:

- ➔ **0** – dann wird die Spalte „laufende Nummer“ nicht mit ausgedruckt
- ➔ **1** – jetzt wird die Spalte „laufende Nummer“ mit ausgedruckt

Achtung: Verwenden Sie nur diese zwei Werte! Andere Zahlen führen zu Fehlfunktionen, die Werte werden jedoch intern nicht mehr geprüft.

Weitere Individualisierung der Berichte ist nur im Code selbst möglich. Hierzu jedoch sollte man sich zunächst wirklich mit dem Programmfluss und der Logik des Programms vertraut machen.

4.2 Dialoge verändern

Alle Dialoge sind im Basic-Dialog Modul vordefiniert und können dort entsprechend verändert werden. Dies betrifft insbesondere die verwendeten Schriften, die Farben, aber auch die Größen der Kontrollelemente und der Dialoge an sich.

Das Programm wurde ursprünglich auf Windows XP geschrieben und hierfür optimiert. Selbstverständlich läuft es auch auf Linux oder Mac Systemen, die Optik könnte aber eventuell ein wenig anders ausfallen. Stellen Sie fest, dass auf Ihrem System die Button-Beschriftungen über die Button-Begrenzung hinausragen, Texte unvollständig dargestellt werden oder sich Kontrollelemente überlappen, so passen Sie diese in den jeweiligen Dialog-Modulen entsprechend an.

4.2.1 Projektdialog

Eine typische Änderung im Projekt-Dialog ist das Deaktivieren des Buttons „automatische Projektnummer“. Hier gibt es zwei Möglichkeiten:

1. Deaktivieren Sie die Option **Aktiviert** in den Eigenschaften des Buttons. Jetzt ist der Button funktionslos, aber noch zu sehen. Der Benutzer kann ihn jedoch nicht aktivieren.
2. Die andere Möglichkeit wäre, den Button auf die Seite 2 zu verschieben. Dazu ändern Sie den Eigenschaftswert **Seite (Step)** auf **2**. Dadurch wird der Button auf diese Ebene verschoben, die im normalen Betrieb nicht zu sehen ist. Auch jetzt ist der Button nicht ansprechbar.



Wichtig:

Löschen Sie keine Kontrollelemente! Diese werden immer in irgendwelchen Codes angesprochen – und das Fehlen eines Elementes führt unweigerlich zu einem Fehler! Verschieben Sie unerwünschte Kontrollelemente auf eine hohe, unsichtbare Ebene (Seite). Dadurch bleiben die Elemente erhalten, sind ansprechbar durch das Programm, aber für den Benutzer unsichtbar und nicht aktivierbar.

5 Dokumentation

Die folgenden Ausführungen erläutern die Arbeitsweise und den Aufbau des Programms. Es wird hier allerdings nicht jede Codezeile einzeln „besprochen“, hierzu muss sich der interessierte Leser schon den Code direkt betrachten. Aufgrund dieser Dokumentation sollte es möglich sein, Änderungen oder Erweiterungen am Programm vorzunehmen.

Der Code selbst beinhaltet viele Kommentare und Kurzerläuterungen, so dass sich auch vieles direkt im Code erkennen lässt.

Das Programm OpenOffice.org Kassenbuch ist vollständig in OpenOffice.org Basic realisiert und man sollte entsprechende Kenntnisse mitbringen, bevor man Änderungen am Code oder am Programm vornimmt.

Die Dialoge sind alle in Dialogmodulen vordefiniert, teilweise auch in mehreren Ebenen. Als Basis für das Programm dient eine Calc-Datei, die wiederum über unterschiedliche Schaltflächen verfügt und damit Makrocode startet.

5.1 Programmfluss

5.1.1 Aufgabenstellung

Das OpenOffice.org Kassenbuch basiert auf folgenden Anforderungen und Nebenbedingungen, wobei insgesamt eine möglichst benutzerfreundliche Bedienung möglich sein soll:

- Das Kassenbuch soll die „einfache“ Buchhaltung einer Barkasse abbilden.
- Belege sollen einfach (über eine Dialogmaske) eingebucht werden können.
- Die Belege sollen einem Konto zugeordnet werden können.
- Belegbuchungen sollen einem Monat (Belegdatum) zugeordnet werden.
- Belege können Einnahmen oder Ausgaben darstellen. Es gibt weiterhin Einnahme- und Ausgabe-Konten. Die Möglichkeit der Zuordnung muss auf die jeweilige Kategorie begrenzt werden.
- Die Monatslisten sollen in Journalform geführt werden. In der Kopfzeile soll ein Übertrag-Betrag des Vormonats auftauchen, am Ende der Liste sollen in Fusszeilen die Summen gebildet werden und der Saldo aus Einnahmen und Ausgaben, der dann gleichzeitig als Übertrag für den nächsten Monat dient.
- Zwischen dem letzten Listeneintrag und der Fusszeile sollen keine Leerzeilen vorhanden sein.
- Die Monatslisten sollen sich einfach ausdrucken lassen.
- Das Kassenbuch soll einen aktuellen Stand der Kasse darstellen.
- Es soll eine Jahresauswertung vorhanden sein, in der die Monate in Spalten und die Konten in Zeilen über das Jahr hin ausgewertet werden. Die Auswertung bildet auch Summen.
- Die Zuordnung der Konten zu der Jahresauswertung soll flexibel gehandhabt werden können.
- Bei bestimmten Sammelkonten soll eine weitere Unterteilung der Zuordnung der Belege möglich sein – also eine Projektverwaltung (oder auch Kostenstelle).
- Konten und Projekte sollen eine Budget-Verwaltung erhalten, das heisst, man kann einen bestimmten Ein- oder Ausgabebetrag vorgeben, intern wird dieser Wert regelmäßig mit dem summierten Werten verglichen.
- Konten und Projekte sollen einfach einzeln ausgewertet werden können – eine Liste zeigt dann alle Buchungen dieses Kontos bzw. diese Projektes an und summiert die Beträge auf. Ein Vergleich zum Budget soll angezeigt werden.
- Neben den Barausgaben soll das Kassenbuch auch Rechnungen – also unbare Belege – erfassen und quasi vorsortieren können. Dazu soll es eine eigene Liste geben. Auch Rechnungen können Konten und Projekten zugeordnet und sollen bei der Auswertung getrennt aufgelistet werden.

- Und schliesslich soll ein einfacher Kassenabgleich möglich sein, also die Erfassung der tatsächlich vorhandenen Barmittel im Vergleich zu den Buchwerten.

Um diese Funktionen sinnvoll nutzen zu können, gibt es auch noch ein paar Nebenbedingungen (hier gilt der Begriff „Beleg“ als Überbegriff für Barbelege und Rechnungen):

- Ein Beleg muss auch nach der Buchung bequem änderbar sein – in all seinen Werten, also Ein-/Ausgabe, Konto, Betrag und Datum.
- Ein Beleg darf nicht mehr löschtbar sein (durch den Benutzer), wohl aber soll er storniert werden können.
- Es muss eine Möglichkeit bestehen, abgeschlossene Monate auch wirklich „abzuschliessen“, das bedeutet, eine Änderung der Belege ist dann nicht mehr möglich.
- Die Ausdrucke / Listen sollen bestimmte Stammdaten mit beinhalten zur einfachen Identifikation.

Diese Anforderungskatalog galt es nun umzusetzen.

5.1.2 Realisierung

Da bei der Realisierung bewusst auf eine Datenbank verzichtet werden sollte und das Konzept auf der Basis von Calc entstehen muss, war die Grundstruktur schnell klar: Pro Monat eine Tabelle, Auswertungen extra.

Die Belegerfassung wird über eine Dialogmaske realisiert, so muss der Benutzer keine Eintragungen direkt in der Tabelle vornehmen. Dadurch sinkt die Fehlerquote und Fehleingaben. Der Dialog erfasst alle wichtigen Daten wie Belegart (Ein- oder Ausgabe), Belegdatum, Konto, Beschreibung, eventuell Projekt, sowie den Betrag.

Neben den Belegen werden Rechnungen in einer ähnlichen Maske erfasst.

Das Eintragen der Daten in die jeweiligen Monatstabellen beziehungsweise Rechnungstabelle übernehmen Makros, ein direkter Benutzer-Eingriff ist nicht vorgesehen. Die Kopfzeilen der Monatstabellen werden fest vorgegeben (auch hinsichtlich der Platzierung), die Fusszeile ebenso, wobei hier die Platzierung je nach Anzahl der Datensätze schwankt. Zum Einfügen eines neuen Datensatzes wird dann eine komplette neue Zeile in die Tabelle eingefügt, so „verrutschen“ die Fusszeilen entsprechend. Die Schwierigkeit besteht nun, die Summen als Formeln auszubilden in der Art, das auf wechselnde Zeilenanzahlen dennoch richtig reagiert wird. Nun könnte man die Formeln jeweils beim Einfügen einer neuen Zeile per Makro neu schreiben, da jedoch die Auswertungen ebenfalls auf die Monate zugreifen, wird hier ein anderer Weg gewählt: Die Formeln werden über benannte Bereiche definiert, die benannten Bereiche (der Bereich der Datenzeilen) werden per Makro jeweils neu fixiert.

Über die benannten Bereich, welche dann je intern auch als Datenbank fungieren, können schliesslich recht einfach die Auswertungen mit Hilfe der Datenbank-Formeln realisiert werden. Um eine Auswertung auch hinsichtlich der einzelnen Konten zu gewährleisten, müssen für jedes Konto Kriterienbereiche eingerichtet werden. Diese Formeln und die Hilfstabelle bietet dem Benutzer keinen direkten Mehrwert und wird daher ausgeblendet, sie wird nur intern benötigt.

Jede Buchung (Beleg oder Rechnung) wird weiterhin per Makro analysiert und der Betrag dem Kontosaldo hinzuaddiert. Insofern gibt es eine Tabelle Konten, welche alle Konten in einer Liste führt und dort auch die summierten Beträge aller Buchungen für das einzelne Konto darstellt. Dadurch bleibt eine Übersicht erhalten.

Gleiches gilt für Projekte.

Eine Auswertung von einzelnen Konten oder Projekten erwartet eine Liste aller Buchungen. Um diesen Zweck zu erreichen, wird ein Makro Buchungszeile für Buchungszeile durchgehen und auf Übereinstimmung zum Konto/ Projekt überprüfen. Besteht eine Übereinstimmung, so wird diese Buchungszeile in die Auswertung kopiert. Es entsteht eine komplexe Einzelauswertung. Um ein spezielles Konto oder Projekt auswählen zu können, bedarf es eines Auswahldialoges, damit der Benutzer einfach seine Wahl treffen kann.

Etwas komplizierter wird es, wenn ein bereits gebuchter Beleg geändert werden soll. Als erstes bedarf es auch hier eines Auswahl-Dialoges, damit der Benutzer den gewünschten Beleg einfach identifizieren kann. Dann wird der Buchungssatz entsprechend in den Bearbeitungsdialog geladen. Nun sind folgende Fälle zu berücksichtigen:

1. Es findet gar keine Änderung statt, der Datensatz wird so zurückgeschrieben – einfachster Fall.
2. Der Datensatz wird hinsichtlich des Datums geändert – jetzt muss dieser aus der aktuellen Monatstabelle entfernt und in die neue Monatstabelle eingefügt werden.
3. Der Datensatz wird hinsichtlich seiner Art (Einnahme/Ausgabe) geändert – jetzt müssen die Kontozuordnungen aufgehoben werden, das bedeutet aber auch, der ursprüngliche Betrag muss vom Kontosaldo des ursprünglichen Kontos/Projekt abgezogen und zum neuen Konto/Projekt hinzugefügt werden.
4. Der Datensatz wird hinsichtlich des Betrages geändert – auch hier muss der ursprüngliche Betrag zunächst von den Kontensalden abgezogen und der neue Betrag hinzuaddiert werden.
5. Eine Kombination aus allen.

Da eine Änderung jederzeit abgebrochen werden kann, muss dann der alte Zustand wieder hergestellt werden. Um einen Beleg/Rechnung zu ändern, wird also zunächst dieser Datensatz zwischengespeichert und erst nach Abschluss der Änderungen diese Daten verworfen. Ansonsten werden die oben genannten Fälle einfach alle mit berücksichtigt.

Die Stornierung eines Beleges hingegen ist recht einfach: Der Betrag wird auf Null gesetzt, wobei intern die gleiche Prozedur abläuft wie bei Änderungen.

Alle auszudruckenden Tabellen erhalten einen eigenen Druckbutton und internen Druckaufbereitung. Dadurch wird ein optimales Ergebnis garantiert.

Der Kassenabgleich wird als einziges Formblatt auch direkt Eingaben zulassen und dient dann als Schema einer Kassenabrechnung.

Die Konzentration auf den Benutzer und die Führung desselben über Dialoge macht es unnötig, irgendwelche Änderungen direkt in den Tabellen vorzunehmen. Damit hier keine Fehleingaben passieren, werden alle Tabellen gesperrt. Um dennoch Anpassungen vornehmen zu können, wurde ein „Administrator-Modus“ realisiert. In diesem Modus sind alle Tabellen sichtbar und alle Sperren aufgehoben. Jetzt können beliebige Änderungen vorgenommen werden. Da jedoch viele Bereiche nicht geändert werden dürfen beziehungsweise die Programmroutinen dort bestimmte Inhalte voraussetzen, sollte man Änderungen nur vornehmen, wenn man genau weiss, was man da tut. Insofern wird der Administrator-Modus durch ein Passwort geschützt.

Für die Steuerung intern wird die Kontonummer genutzt. Diese muss numerisch und maximal 4-

stellig sein, also im Bereich von 0 und 9999, wobei die 9999 intern genutzt wird und somit nicht mehr zur Verfügung steht. Neben der Kontonummer wird auch die Projektnummer genutzt, ebenfalls vierstellig.

Damit sind die Grundlagen des Programms erläutert, bleibt noch folgendes anzumerken: Das Kassenbuch kann immer nur für ein Geschäftsjahr genutzt werden (12 Monate), wobei eine unterjährige Führung nicht vorgesehen ist.

5.2 Struktur der Datei

Die Calc-Datei beinhaltet insgesamt 21 Tabellen, die sich folgt darstellen (die Reihenfolge entspricht der Originalversion, diese kann aber verändert werden):

Tabellenname	Beschreibung
Stammdaten	Ist die Hauptseite des Programms. Auf dieser Seite befinden sich die Steuerbuttons für den Benutzer, aufgeteilt in einen Aufgaben- und einen Verwaltungsbereich. Gleichzeitig bietet die Seite einen Überblick über die Stammdaten, den aktuellen Kassenstand sowie Lizenzkurzhinweise. Scrollt man die Tabelle nach rechts, so erscheinen zusätzlich die internen Daten wie Nummernkreise und mehr. Scrollt man weiter, erscheint eine Einnahme/Ausgaben Übersicht für die jeweiligen Monate. Diese Tabelle ist eine Zwischentabelle welche intern genutzt wird, sie ist aber auch informativ.
Projekte	Liste aller Projekte, pro Zeile ein Projekt. Die Projektseite zeigt auch für jedes Projekt einen aktuellen Status, neben einem eventuellen Budget die aufgelaufenen Summen sowie die Budget-Erfüllung in Prozent und ein eventuelles Restbudget. Die Seite bietet Möglichkeiten, Projekte hinzuzufügen, diese zu verwalten, auszuwerten oder die Projektsalden zu aktualisieren. Hierbei werden alle Projektsalden neu berechnet – das kann einige Zeit dauern.
Konten	Liste aller Konten, pro Zeile ein Konto. Die Kontotabelle zeigt auch für jedes Konto einen aktuellen Status, neben einem eventuellen Budget die aufgelaufenen Summen sowie die Budget-Erfüllung in Prozent und ein eventuelles Restbudget. Die Seite bietet Möglichkeiten, Konten hinzuzufügen, diese zu verwalten, auszuwerten oder die Kontensalden zu aktualisieren. Hierbei werden alle Projektsalden neu berechnet – das kann einige Zeit dauern. Die Reihenfolge der Konten ist beliebig – auch wenn in der Originalversion zuerst die Einnahme und dann die Ausgabenkonten folgen.

Formel	<p>Diese Tabelle ist nur im Administratormodus sichtbar. Sie beinhaltet ein Administrator-Menü, das Passwort und das Datum der Erstinitalisierung. Des weiteren stehen hier Formeln und Bereich, die intern genutzt werden. Diese werden überwiegend per Makro geschrieben, ein manueller Eingriff ist weder sinnvoll noch empfehlenswert.</p> <p>In einem weitem Bereich befinden sich Zwischenspeicher. Dort werden die Daten eines zu ändernden Beleges oder einer Rechnung zwischengespeichert.</p>
Januar bis Dezember	<p>Diese 12 Tabellen stellen die jeweiligen Monatsjournale dar. Hier finden sich später alle (Beleg-) Buchungen wieder. Jede Seite enthält einen benannten Datenbereich (Jan bis Dez mit Ausnahme MaiDB), welcher sowohl für die Summenformeln auf den Seiten als auch in den Auswertungen genutzt wird.</p>
lfd_Rechnungen	<p>Dieses Journal ist ähnlich den Monatsjournalen aufgebaut und nimmt alle Rechnungen als Datensatz auf. Auch diese Tabelle trägt einen benannten Bereich (RechDB), der dann wiederum für die Formeln bzw. Berechnungen genutzt wird.</p>
Auswertungen	<p>Die Auswertungstabelle enthält eine dynamisch angepasste Auswertungstabellen, und zwar werden die Summen in Monatsspalten und Kontenzeilen dargestellt, aufgeteilt nach Einnahmen und Ausgaben. Zusätzlich gibt es Zeilen- und Spaltensalden.</p> <p>Inwieweit ein Konto in den Auswertungen aufgenommen wird, kann über einen Schalter beim Konto selbst bestimmt werden.</p>
Projektkosten	<p>Die Tabelle Projektkosten ist Träger eines Berichtes für die Auswertung einzelner Projekte. Dabei werden alle Buchungen für ein Projekt chronologisch aufgelistet, die Summe gebildet und ein Vergleich zum eventuell bestehenden Budget angestellt. Diese Seite ist nicht dynamisch und wird immer aktuell erstellt, wenn ein neues Projekt ausgewählt oder der entsprechende Button gedrückt wird.</p>
Kontenauswertung	<p>Die Tabelle Kontenauswertung ist Träger eines Berichtes für die Auswertung einzelner Konten. Dabei werden alle Buchungen für ein Konto chronologisch aufgelistet, die Summe gebildet und ein Vergleich zum eventuell bestehenden Budget angestellt. Diese Seite ist nicht dynamisch und wird immer aktuell erstellt, wenn ein neues Konto ausgewählt oder der entsprechende Button gedrückt wird.</p>
Kassenabgleich	<p>Die Tabelle Kassenabgleich ermöglicht einen einfachen Vergleich des Kassenbestandes lauf Buchführung und dem Ergebnis einer direkten Zählung des tatsächlichen Bestandes. Über die einzelnen Kategorien lassen sich die gezählten Bestände leicht und schnell erfassen.</p>

Alle Tabellen verfügen über eingebundenen Schalter, die einmal eine Schnellnavigation zwischen den Tabellen ermöglichen, auf der anderen Seite aber auch typische Aktionen der aktuellen

Tabellen starten, so wie „drucken“ oder „Auswahl“. Aufgrund der Schalter ist es dem Benutzer möglich, das Programm völlig ohne Eingriff in die Tabellen zu steuern und alle gewünschten Aktionen durchzuführen.

Die folgenden Darstellungen beschreiben die den Schaltern zugewiesenen Makroprogramme und geben Kurzbeschreibungen der Funktionsweise.

5.2.1 Start der Datei

Schon beim Start der Datei wird ein Makro ausgeführt, und zwar die Routine **prog_start** im Modul **Init_Mod**. Dieses ist sowohl mit dem Ereignis **Dokument erzeugen** als auch mit dem Ereignis **Dokument öffnen** verbunden. Wichtig ist das erste Ereignis immer dann, wenn aus der Vorlage ein neues Kassenbuch erzeugt wird, das zweite ist selbsterklärend.

Diese Routine hat folgende Aufgaben: Erstens werden interne Variablen initialisiert, dann wird der Administratormodus ausgeschaltet – falls er beim letzten Speichern und Schliessen der Datei noch eingeschaltet war. die Stammdatenseite wird als die sichtbare Starttabelle initialisiert und schliesslich wird beim ersten Start der Datei der Erstinitialisierungsdialog aufgerufen. Gesteuert wird dieses Ereignis über den Eintrag in der Zelle **B3** in der Tabelle Formeln. Steht dort bereits ein Datum drin, wird der Dialog nicht aufgerufen, sonst ja.

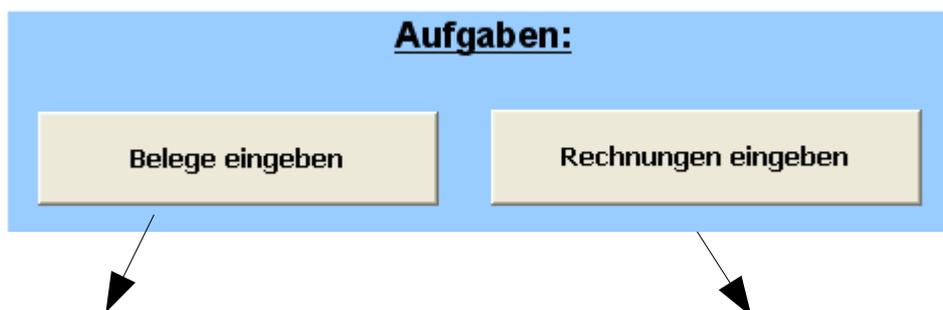
```
REM Funktion ist mit dem Programmstart verbunden
sub prog_start
  init
  REM Evt. Admin Modus austellen
  change_AdminModus(false) 'modul Init_Mod
  but_geheStart           'Stammdatenseite aufrufen
  if oForTab.getCellRangeByName("B13").value = 0 then
    REM Erststart des Programms
    Start_initDlg(true)
    speichern() 'Dokument speichern - Modul Helper
  end if
end sub
```

Die Funktion **but_geheStart** des Moduls Main legt die Tabelle „Stammdaten“ als sichtbare (aktive) Tabelle fest, auch wenn sie an eine andere Stelle als die erste Tabelle verschoben wurde.

5.2.2 Tabelle Stammdaten

Die Tabelle **Stammdaten** ist die Haupttabelle und beinhaltet neben den Stammdaten auch jede Menge Optionen sowie die kompletten Steuerelemente.

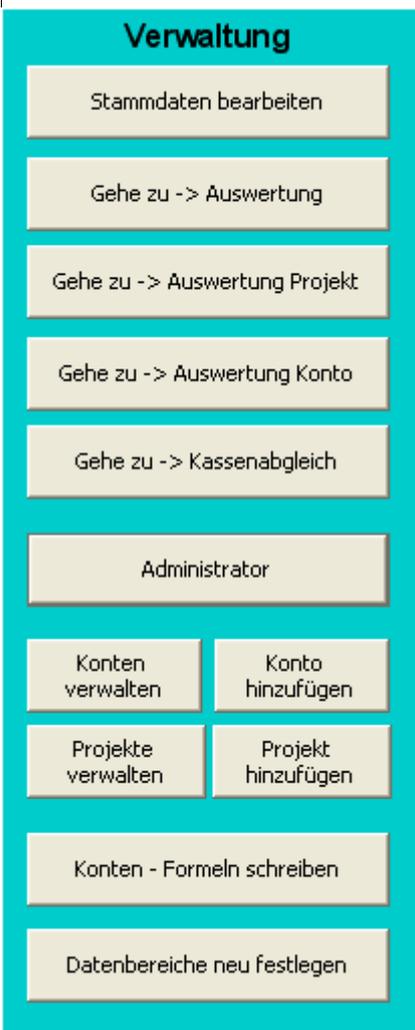
Die folgenden Darstellungen dokumentiert die Verknüpfung der Schaltelemente zu den entsprechenden Makros. Dabei erfolgt die Darstellung in der Form: Modul.Makro



verbunden mit Main.But_Belege_eingeben -> öffnet den Dialog Belege eingeben.	verbunden mit Main.But_Rechnungen_eingeben -> öffnet den Dialog Rechnungen eingeben.
----------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------

Beide Buttons rufen den gleichen Dialog auf (Beleg), jedoch in unterschiedlichen Stufen (Seiten/Steps).

Die Verwaltungsbuttons:

	<p>Die Buttons sind verbunden mit: (Modul.Makro)</p> <p>Init_Mod.Stammdaten - öffnet den Dialog Stammdaten</p> <p>Main.but_geheAuswertung</p> <p>Main.but_gehePrjAuswertung</p> <p>Main.but_geheKtoAuswertung</p> <p>Main.but_geheKasseAuswertung</p> <p>Admin.Admin_modus - schaltet den Admin-Modus ein, ruft zunächst den Passwort-Dialog auf.</p> <p>Konten.Kontoauswahl Konten.KontoHinzufuegen – ruft die entsprechenden Dialoge auf.</p> <p>Projekte.Projektauswahl Projekte.np_ausTabelle</p> <p>Konten.KontenBereicheSchreiben – schreibt alle Kontenformeln neu, die wiederum für die Auswertung benötigt werden.</p> <p>Admin.Bereiche_definieren – Die Datenbereiche (benannte Bereiche) werden neu definiert. Wichtig für Formeln und Auswertung.</p>
------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Die Gehe-zu Buttons springen lediglich zu den entsprechenden Tabellen – das erleichtert die Navigation.

Weitere Elemente der Tabelle Stammdaten:

In der Spalte C in den Zeilen 6-11 stehen die Stammdaten – Daten, die über den Stammdaten-Dialog eingegeben und geändert werden können. Wichtig sind dabei:

6	Geschäftsjahr:	2006
7	Einrichtung:	JugZ gGmbH Köln
8	Kostenstelle:	KSt. 25 - Herr Mustermann
9	Adresse:	teststr. 4
10		23456 Köln
11	Kontakt:	Tel. 012345-505005

Das **Geschäftsjahr** – dieses dient auch als Kopfzeile in den diversen Berichten.

Einrichtung: Hier steht normalerweise der Name der Einrichtung/Organisation

Kostenstelle: Dies ist eine Zusatzinformation zur Einrichtung,

kann auch eine Zweigstelle, eine bestimmte Bezeichnung, eine Kostenstelle, ein Ansprechpartner oder was auch immer sein.

Wichtig: Die Berichte bekommen Kopfzeilen bestehend aus dem Inhalt der Zellen **C7 + C8**, beide Zellinhalte werden verknüpft und auf den Bericht gedruckt. Die Kopfzeile wird per Makro erzeugt!

Ein weiterer, interessanter Bereich sind die Optionen:

G	H
	Optionen:
	Belegbeschreibungen /Firmen
Fest	Tankbeleg
	Büromaterial
	Rechnung
	Software
	Hardware
	Werkzeug / Maschinen
Variabel	Plan GmbH
	Test

Diese beginnen ab Spalte **C**. In den Zeilen **4 bis 29** stehen die Einträge der Listbox Firma/Beschreibungen in der Belegerfassungsmaske. Dabei werden die ersten sechs Einträge immer verwendet (diese sind fest und vom Administrator hier änderbar) , die restlichen 20 Einträge dynamisch durch die Combobox „gefüllt“, das bedeutet, wählt der Benutzer keinen Begriff aus der Liste aus sonder schreibt einen neuen in die Eingabezeile, so wird dieser als 7. Begriff hier in Zelle **H10** eingetragen. Alle weiteren Einträge wandern somit eine Stufe nach unten, wobei der letzte Eintrag (bisherige **H29**) entfällt.

Natürlich kann der Administrator die Liste hier auch manuell auffüllen, allerdings unterliegen auch dann die Einträge in den Zellen H10 bis H29 dem geschilderten Rotationsprinzip. Lediglich die Einträge in H4 bis H9 werden immer angezeigt und nicht geändert.

Weitere Optionen befinden sich dann in der Spalte L, wobei die Erläuterungen in Spalte K zu finden sind. Auch die meisten dieser Optionen sind über den Stammdaten-Dialog einzustellen.

Lediglich die laufende Nummer kann nicht

eingestellt werden. Dies ist allerdings auch nicht sinnvoll. Die Werte werden jeweils bei Buchungen per Makro angepasst und zurückgeschrieben. Dies gilt übrigens auch für alle andern links dargestellten Werte. Ein manueller Eingriff ist meist nicht sinnvoll und sollte unterbleiben.

Letzte laufende Nummer:	36
Letztes Eingabedatum:	16.09.2006
Letzte Belegnummer:	2006-028
Letzte Rechnungsnummer:	RE-2006-008
Letzte Preislistennummer:	0004
Linkes Logo:	jugz.tif
Rechtes Logo:	katze.jpg

Rechnungs- und Belegnummern werden intern als „Strings“ behandelt und entsprechend so auch hier gespeichert. Es können beliebige alphanumerische Zeichenfolgen sein, diese dürfen jedoch keine Leerzeichen enthalten. Der jeweils letzte Nummernblock wird dann intern hochgezählt (vergleiche auch **Helper.NumAdd**).

Neben den Nummernkreisen sowie den Zwischenspeicher für das Datum finden sich weiterhin diverse Optionsfelder auf der Seite Stammdaten. Da wären zunächst die beiden Logos, die in der Belegmaske eingebettet werden können. Diese werden in den Zellen **L19** und **L20**

spezifiziert und während der Initialisierung ausgelesen. Wird die Belegmaske dann aufgerufen, erhalten die vorhandenen Grafikfelder die URL der Logos zugewiesen, und zwar eine Kombination aus dem aktuellen Pfad der Programmdatei und den hier in den Zellen hinterlegten Dateinamen – eventuell mit weiteren Pfadangaben versehen.

Auch die Optionen zum Sperren der Belege findet sich in der Tabelle Stammdaten, und zwar in den Zellbereichen **L23 –**

Zurückliegende Monate sperren:	1	1 = ja, 0 = Nein
Monate zurück:	2	1 – 11
letzter gesperrter Monat:	6	1 – 12

L26. Die dort befindlichen Daten werden von den diversen Makros eingelesen und ausgewertet.

E
aktueller Kassenstand vom 08.10.2006
295,00 €

Und schliesslich befindet sich eine Tabelle mit Zwischenwerten in der Stammdaten-Tabelle. Diese Werte werden benötigt zur aktuellen Anzeige des Kassenstandes, wie er in Zelle **E2** dargestellt wird. Im Grunde beinhaltet die Zelle lediglich eine Referenz auf die Zelle **R19**, die Summe der

einzelnen Monatssalden. Die Saldenverwaltung selbst wird dynamisch über die benannten Bereiche der einzelnen Monate und entsprechende Datenbank-Formeln aktualisiert.

Dabei werden jeweils die Ein- und Ausgaben unabhängig von einander ausgewertet (die entsprechenden Kriterienbereiche finden sich in der Tabellen **Formeln**) und die Einzelwerte über Summenformeln hier addiert. Daraus ergibt sich dann der aktuelle Kassenstand (unten rechts als Summe der Monatssalden – und zur Kontrolle als Summe der Einnahmen abzüglich der Ausgaben.

O	P	Q	R
Kassenbuch – Übersicht:			
	Einnahmen	Ausgaben	Saldo
Anfangssaldo:			0,00 €
Januar	0,00 €	0,00 €	0,00 €
Februar	0,00 €	5,00 €	-5,00 €
März	0,00 €	4,00 €	-4,00 €
April	0,00 €	0,00 €	0,00 €
Mai	0,00 €	12,00 €	-12,00 €
Juni	0,00 €	0,00 €	0,00 €
Juli	0,00 €	0,00 €	0,00 €
August	602,00 €	497,00 €	105,00 €
September	276,00 €	65,00 €	211,00 €
Oktober	0,00 €	0,00 €	0,00 €
November	0,00 €	0,00 €	0,00 €
Dezember	0,00 €	0,00 €	0,00 €
Summen:	878,00 €	583,00 €	295,00 €

Eine Formel zur Berechnung der Einnahmen oder Ausgaben sieht beispielhaft wie folgt aus (Einnahmen für den Monat Januar – Zelle P6):

```
=DBSUMME(Jan;
"Einnahmen";
$Formeln.$F$3:$F$4)
```

Damit wäre die Struktur der Tabelle Stammdaten erläutert.

5.2.3 Tabelle Formeln

Die Tabelle Formeln ist für den normalen Benutzer nicht sichtbar. Sie trägt neben den Passwort und dem Datum der Erstinitialisierung Formeln für die Auswertung und für die Datenbereiche. Intern wird die Auswertung regelmäßig über

benannte Bereiche und dann über Datenbank-Formeln erreicht. Diese Datenbank-Formeln erfordern in der Regel drei Parameter:

Den Datenbankbereich – leicht zu lösen über den Namen des benannten Bereiches
 das Suchkriterium (Spaltenname der auszuwertenden Spalte) – in der Regel direkt eingetragbar
 den Kriterienbereich – dort, wo die Kriterien stehen, welche Daten ausgewertet werden.

Dieser Kriterienbereich wird in der Tabelle Formeln festgeschrieben.

Benötigt werden folgende Kriterienbereiche:

17	Ab Zeile 20 finden sich für jedes Konto entsprechende	
18	<u>Kriterienbereiche</u> – Für die Auswertung:	
19		
20	1658	Konto
21	1658	1658
22	1658	
23	5500	Konto
24	5500	5500
25	5500	
26	5701	Konto
27	5701	5701
28	5701	
29	5510	Konto
30	5510	5510
31	5510	
32	5530	Konto
33	5530	5530
34	5530	

Für die Auswertung muss jedes Konto einen eigenen Kriterienbereich haben – zum Vergleich mit der Kontonummer. Dieser beginnt ab Zeile 20 und zwar in der Form, dass in Spalte A die Kontonummer aufgeführt wird und in Spalte B der eigentliche Kriterienbereich. Benötigt werden immer drei Zeilen pro Konto: Die erste Zeile des Kriterienbereichs trägt die Spaltenüberschrift (hier immer „Konto“), die zweite Zeile beinhaltet das Suchkriterium, bei deren Übereinstimmung der Datensatz ausgewertet wird – in

diesem Fall die jeweilige Kontonummer – und die dritte Zeile ist leer, zur Abgrenzung, dass der

Kriterienbereich jetzt hier aufhört und keine „Oder“ Bedingung folgt. Die erste Spalte (A) dient zur Information und zeigt einfach an, welcher Kriterienbereich zu welchem Konto gehört.

Wichtig:

Nehmen Sie hier keine manuellen Änderungen vor. Die Kriterienbereiche werden über ein internes Makro geschrieben – und die jeweilige Position für das benannte Konto in der **Spalte G** der Tabelle **Konten** in der Kontozeile abgespeichert. Die Spalte G ist in der Regel ausgeblendet und somit nicht sichtbar, auf die wird aber per Makro zugegriffen zum Finden des Kriterienbereichs!

Ein weiterer wichtiger Kriterienbereich findet sich in der Spalte F und G der Tabelle Formeln, ebenfalls ab Zeile 20 bis Zeile 54. Hier werden die einzelnen Monate als Kriterium hinterlegt – zur Auswertung der Monatsalden. Dieser Bereich steht fest und darf nicht verändert werden, internen Formeln und Makros greifen hierauf zu.

Der Bereich ist wie folgt aufgebaut:

<u>Kriterienbereiche für die Auswertung der Rechnungen</u>		
Januar	Datum	Datum
	>= 01.01.	< 01.02.
Februar	Datum	Datum
	>= 01.02.	< 01.03.
März	Datum	Datum
	>= 01.03.	< 01.04.
April	Datum	Datum

In Spalte **E** stehen die Monatsnamen – nur zur Information. Eine Auswertung oder interne Nutzung erfolgt nicht.

Auch hier ist der Kriterienbereich dreizeilig: die erste Zeile beinhaltet den Spaltennamen, hier sowohl in Spalte **F** als auch in Spalte **G** den Begriff **Datum**.

In der nächsten Zeile sind dann die Bedingungen definiert, wobei

die beiden neben einander liegenden Spalten als „Und“-Verknüpfung verstanden werden, also beide Bedingungen müssen zutreffen. Zu finden sind dort also immer das Anfangs- und Enddatum eines Monats. Der Wert „< Startwert des nächsten Monats“ sichert hierbei auch die Berücksichtigung von Schaltjahren, das heißt unbestimmten Monatsenden.

Die dritte Zeile schliesslich ist wieder leer zur Abgrenzung des Kriterienbereiches.

Ein letzter Kriterienbereich befindet sich in der **Spalte A** bis **I** in den **Zeilen 3** bis **5**. Dort wird die Struktur der benannten Bereiche in den Monatstabellen wiederholt ohne einzelne Suchkriterien einzustellen. Die Zeile 4 und 5 müssen auch leer bleiben, sonst würden nicht mehr alle Datensätze gefunden.

	A	B
9		
10	Passwort:	
11		
12		
13	<u>Erstinizialisierung:</u>	09.08.2006
14		

Neben den Kriterienbereichen enthält die Tabelle Formeln noch – wie bereits erwähnt – das Passwort sowie das Datum der Erstinitialisierung. Dieses Datum wird in die Zelle B13 beim Ersten Aufruf des Programms eingetragen – aus dem Systemdatum und steuert später, ob der Initialisierungsdialog aufgerufen wird oder nicht. Das Passwort selbst wird in roter

Schrift formatiert, so dass es normalerweise nicht zu sehen ist. Lediglich in der Eingabezeile kann man es schliesslich lesen.

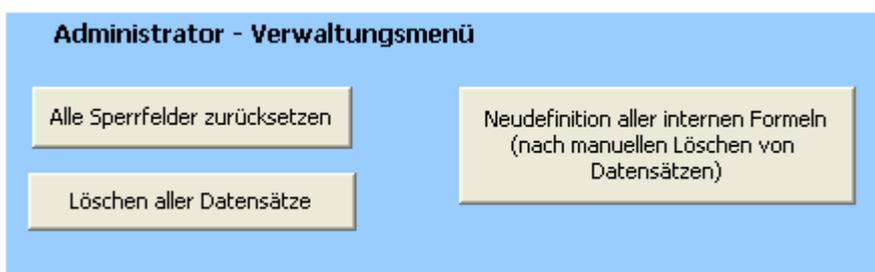
Dann enthält die Tabelle Formeln noch einen Zwischenspeicher. Wird ein Beleg oder eine Rechnung geändert, so werden die ursprünglichen Daten erst einmal zwischengelagert, um diese im Falle eines Dialogabbruchs rekonstruieren und wiederherstellen zu können.

	P	Q	R
1	Zwischenspeicher Projektdaten:		
2	ProjNr	Einnahme	Ausgabe
3	0	0	0
4			
5	Zwischenspeicher Kontodaten:		
6	KontoNr	Einnahme	Ausgabe
7		0	0
8			

Diese Zwischenspeicher finden sich in den **Spalten P,Q und R** und in den Zeilen **1-7**, wobei nur die **Zeile 3 und 7** tatsächlich von Makros genutzt werden, der Rest ist einfach Information.

Daten werden dort per Makro eingetragen und auch wieder gelöscht – eine manuelle Änderung würde nichts bringen, vorhandene Daten werden überschrieben.

Zum Schluss noch die Verknüpfungen des Verwaltungsmenüs, welches sich in den Zellen C6 bis G15 befindet:



<p>Alle Sperrfelder zurücksetzen ist verbunden mit: Admin.Sperren_aufheben</p>	<p>Neudefinition aller internen Formeln ist verbunden mit: Admin.NeuBerechnenenIntereFormeln</p>
<p>Löschen aller Datensätze ist verbunden mit: Admin.But_del_all</p>	

5.2.4 Die Tabelle Konten:

Die Kontentabelle besitzt vier Schaltknöpfe ganz oben, denen folgende Makros verbunden sind:



verbunden mit:

Konten.KontoHinzufuegen	Konten.Kontoauswahl	Konten.KontenSaldenNeuSchreiben	Konten.Kontoauswah
-------------------------	---------------------	---------------------------------	--------------------

Ansonsten ist die Tabelle wie folgt aufgebaut:

Spalte B: Art des Kontos. Die Spalte ist bedingt formatiert, je nach Inhalt wird der Zellhintergrund rot (Inhalt **A** wie Ausgabe) oder grün (Inhalt **E** wie Einnahme).

B	C	D	E	F	G	H	I	J	K
Art	Kontonummer	Kontobeschreibung	Aktiv	Auswertung	Bereich	Budget	Summen	In_Prozent	Restbudget
E	1658	diff. Schecks (Einnahmen)	1	0	\$Formeln.\$B\$20:\$B\$21	125.000,00 €	0,00 €	0,00%	125.000,00 €
E	5500	Eigeneinnahme Projekte	1	1	\$Formeln.\$B\$23:\$B\$24		690,00 €		
E	5701	Erlöse Raumvermietung	1	1	\$Formeln.\$B\$26:\$B\$27		0,00 €		
E	5510	Erlöse Veranstaltungen	1	1	\$Formeln.\$B\$29:\$B\$30		0,00 €		
E	5530	Erlöse aus Verzehr	1	1	\$Formeln.\$B\$32:\$B\$33		182,00 €		
E	5750	Erlöse Telefon	1	1	\$Formeln.\$B\$35:\$B\$36		45,00 €		
E	5705	Erlöse aus Vermietung sttr.	1	1	\$Formeln.\$B\$38:\$B\$39		0,00 €		
E	5505	Elternbeiträge	1	1	\$Formeln.\$B\$41:\$B\$42		0,00 €		
E	5550	Erlöse Essensgelder ÜMB	1	1	\$Formeln.\$B\$44:\$B\$45		0,00 €		

Spalte C: Kontonummer. Numerischer, maximal vierstelliger Wert. Spalte ist formatiert mit Zahlencode „0000“, führende Nullen werden also angezeigt.

Spalte D Kontobeschreibung

Spalte E: Konto ist aktiv (1) oder nicht (0). Nicht aktive Konten werden in der Listbox des Belegdialoges nicht angezeigt.

Spalte F: Konto wird mit ausgewertet in der Tabelle Auswertung – Ja (1) oder Nein (0). Steht der Schalter auf „Nein“, dann wird das Konto in der Auswertungsübersicht nicht angezeigt.

Spalte G: (normalerweise ausgeblendet) – der Bereich, in dem der Kriterienbereich dieses Kontos zu finden ist.

Spalte H: eventueller Budget-Betrag

Spalte I: Bisher aufsummierte Summen des Kontos. Die Summe wird durch die Makros beim Buchen von Belegen angepasst.

Spalte J: Die Summe in Spalte I entspricht in Prozent des Budgets. Die Formel in dieser Zelle lautet: `=WENN(H7; I7/H7; "");`

Spalte K: eventuelles Restbudget. Die Formel lautet: `=WENN(H7; H7-I7; "");`

Die Kontenliste wird in **Spalte C** (Kontennummer) regelmäßig durch Makros ausgewertet. Die Schleifen enden beim Auftreten einer leeren Zelle. Alle weiteren später folgenden Informationen werden ignoriert.

5.2.5 Die Tabelle Projekte

Ähnlich wie die Konten ist die Tabelle Projekte aufgebaut, allerdings gibt es nur drei Schalter am Anfang:



verbunden mit:

Projekte.np_ausTabelle	Projekte.ProjektSaldenNeuSchreiben	Projekte.Projektauswahl
------------------------	------------------------------------	-------------------------

Ansonsten ist die Tabelle wie folgt aufgebaut:

	A	B	C	D	E	F	G	H	I
6	Projekt_Nr.	Projektbeschreibung	Budget	Einnahmen	Ausgaben	In_Prozent	Restbudget	Typ	Aktiv
7	0001	mein 1. Projekt	500,00 €	150,00 €	327,00 €	35,40%	323,00 €	Ausgabe	1
8	0002	testprojekt 2	1.000,00 €	440,00 €	0,00 €	44,00%	560,00 €	Einnahme	1
9	0003	test3	2.500,00 €	0,00 €	155,00 €	6,20%	2.345,00 €	Ausgabe	1
10	0006	test_kk	150,00 €	0,00 €	0,00 €	0,00%	150,00 €	Einnahme	1
11	0025	Einnahme-Test	1.000,00 €	105,00 €	0,00 €	10,50%	895,00 €	Einnahme	1

Spalte A: Projektnummer. Numerischer, maximal vierstelliger Wert. Spalte ist formatiert mit Zahlencode „0000“, führende Nullen werden also angezeigt.

Spalte B: Projektbeschreibung

Spalte C: Projektbudget

Spalte D: kummulierte Einnahmen des Projektes. Die Werte werden durch die Makros beim Buchen eines Beleges oder einer Rechnung entsprechend angepasst.

Spalte E: kummulierte Ausgaben des Projektes. Die Werte werden durch die Makros beim Buchen eines Beleges oder einer Rechnung entsprechend angepasst.

Spalte F: Die Summe in Spalte I entspricht in Prozent des Budgets. Die Formel in dieser Zelle lautet: $=((-1)* D7 - (-1)* E7) / C7$

Spalte G: eventuelles Restbudget. Die Formel lautet: $=C7 - (-1)* D7 + (-1)* E7$

Spalte H: Typ des Projektes in Klartext (Einnahme oder Ausgabe – andere Texte werden ignoriert)

Spalte I: Aktiv-Schalter. Projekt ist aktiv (1) oder nicht (0). Nicht aktive Projekte werden in der Listbox des Belegdialoges nicht angezeigt.

Die Projektliste wird in **Spalte A** (Projektnummer) regelmäßig durch Makros ausgewertet. Die Schleifen enden beim Auftreten einer leeren Zelle. Alle weiteren später folgenden Informationen werden ignoriert.

5.2.6 Monatstabelle Januar bis Dezember

Die Monatstabellen sind alle gleich aufgebaut. Sie alle tragen im oberen Teil drei Schalter, die wie folgt mit den Makros verbunden sind: Übrigens für alle Tabellen gleich – eine Unterscheidung der aktiven Tabelle wird im Makro selbst gefällt.



verbunden mit:

Main.but_geheStart	Helper.Journal_drucken	Belege_bearb.But_DatVerw
--------------------	------------------------	--------------------------

Ansonsten sind die Tabellen wie folgt aufgebaut:

In **Zelle F1 / G1** stehen eventuelle Monatsüberträge. Die Formeln sehen wie folgt aus (Beispiel Einnahmen Februar): $=WENN(\$Januar.\$F\$10; \$Januar.\$F\$10; "")$

In der **Zelle B2** wird der Monatsname mit dem Geschäftsjahr aus den Stammdaten übertragen. Diese Informationen werden durch die Buttons überdeckt, tauchen dann aber im Ausdruck auf. Die Formel lautet: $=VERKETTEN("Kassenbuch Februar"; " "; \$Stammdaten.\$C\$6)$

In der **Zelle I1** steht ein Schalterwert: 0 oder nichts für „noch nicht erstmalig gedruckt, beim Druck daher Sperrdialog aufrufen“, 1 für „schon erstmalig gedruckt, daher kein Sperrdialog mehr“.

Die Daten stehen dann ab Zeile 4, wobei die Bewegungsdaten ab Zeile 5 beginnen:

	A	B	C	D	E	F	G	H	I
4	Konto	Konto_Bezeichnung	Belegnr.	Datum	Beschreibung	Einnahmen	Ausgaben	Proj-Nr.	Lfd_Nr.
5	6620	Projekte	2006-016	09.02.2006	Test		5,00 €	0000	24
6					Saldo-Vortrag:				
7		Datum:			Einnahmen / Ausgaben:	0,00 €	5,00 €		
8					Gesamt:	0,00 €	5,00 €		
9		Unterschrift:							
10					Saldo-Vortrag für neuen Monat:		5,00 €		
11									

Spalte A: Kontonummer, Numerischer, maximal vierstelliger Wert. Spalte ist formatiert mit

Zahlencode „0000“, führende Nullen werden also angezeigt.

Spalte B: Kontobeschreibung

Spalte C: Belegnummer

Spalte D: Belegdatum

Spalte E: Belegbeschreibung

Spalte F: Einnahmebetrag

Spalte G: Ausgabebetrag

Pro Zeile kann es nur eine Einnahme oder Ausgabe geben. Die Entscheidung wird über das Konto gefällt und vom Makro automatisch richtig gesetzt.

Spalte H: Projektnummer

Spalte I: Laufende Nummer

Die **Saldozeilen** beinhalten Formeln und „wandern“ nach unten. Beispielhaft hier für den Monat Februar, Spalte Einnahmen:

Die Formel des **Saldovortrages** sieht wie folgt aus: `=F$1`

Die Summenformeln (Einnahme) wie folgt: `=DBSUMME(Feb; "Einnahmen"; $Formeln.$F$3:$F$4)`

Die Formel für den Gesamtsaldo dann wie folgt:

`=SUMME(F1;DBSUMME(Feb; "Einnahmen"; $Formeln.$F$3:$F$4))`

Und die Formel für den Saldovortrag für den nächsten Monat dann:

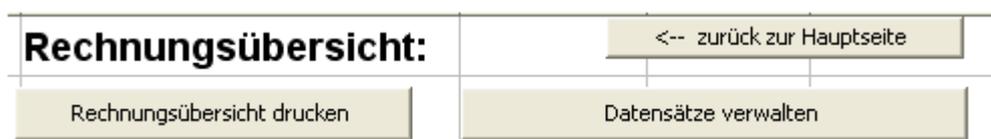
`=WENN(SUMME(F1;DBSUMME(Feb; "Einnahmen"; $Formeln.$F$3:$F$4))-SUMME($G$1;DBSUMME(Feb; "Ausgaben"; $Formeln.$G$3:$G$4))>0;SUMME($F$1;DBSUMME(Feb; "Einnahmen"; $Formeln.$F$3:$F$4))-SUMME($G$1;DBSUMME(Feb; "Ausgaben"; $Formeln.$G$3:$G$4));"")`

Alle Tabellen beinhalten einen benannten Bereich, welcher die Bewegungsdaten pro Monat abbildet. Dieser benannte Bereich wird beim Einfügen eines neuen Datensatzes per Makro automatisch angepasst. Passend zu den Tabellennamen (Monatsnamen) beissen die benannten Bereiche: Jan, Feb, Mar, Apr, MaiDB, Jun, Jul, Aug, Sep, Okt, Nov, Dez (Anmerkung: „Mai“ führt als Name intern zu einem Fehler, daher wurde „MaiDB“ gewählt!)

5.2.7 Tabelle lfd_Rechnungen

Die Tabelle lfd_Rechnungen nimmt ähnlich wie die Monatstabellen einzelne Datensätze auf, jedoch jetzt alle „Rechnungen“ chronologisch und nicht mehr nach Datum getrennt. Im Grunde jedoch ist die Tabelle ähnlich aufgebaut.

Im oberen Bereich befinden sich wieder drei Schaltflächen:



Diese sind wie folgt verbunden:

Helper.Journal_drucken	Main.but_geheStart
	Rechnungen.But_rechnungsverwaltung

Ansonsten ist die Tabelle wie folgt aufgebaut:

Die Daten stehen ab Zeile 4, wobei die Bewegungsdaten ab Zeile 5 beginnen:

Spalte A: Kontonummer, Numerischer, maximal vierstelliger Wert. Spalte ist formatiert mit Zahlencode „0000“, führende Nullen werden also angezeigt.

	A	B	C	D	E	F	G	H	I
4	Konto	Konto_Bezeichnung	Belegnr	Datum	Beschreibung	Einnahmen	Ausgaben	Projekt_Nr	Lfd_Nr
5	6620	Projekte	RE-2006-001	17.08.2006	Büromaterial		25,00 €	0001	3
6	5500	Eigeneinnahme Projekte	RE-2006-002	27.08.2006	Test	40,00 €		0001	13
7	5500	Eigeneinnahme Projekte	RE-2006-003	28.08.2006	Test	100,00 €		0001	14
8	5500	Eigeneinnahme Projekte	RE-2006-004	29.08.2006	Test	0,00 €		0001	15
9	5500	Eigeneinnahme Projekte	RE-2006-005	30.08.2006	PLan GmbH	10,00 €		0001	16
10	6620	Projekte	RE-2006-006	31.08.2006	Tankbeleg		10,00 €	0001	17
11	6620	Projekte	RE-2006-007	23.08.2006	Tankbeleg		5,00 €	0003	18
12	6611	Cyber JugZ	RE-2006-008	31.08.2006	Test		10,00 €		22
13	5550	Erlöse Essensgelder ÜMB	RE-2006-009	09.10.2006	Test	12,20 €			37
14					Summe:	162,20 €	50,00 €		

Spalte B: Kontobeschreibung

Spalte C: Belegnummer

Spalte D: Belegdatum

Spalte E: Belegbeschreibung

Spalte F: Einnahmebetrag

Spalte G: Ausgabebetrag

Pro Zeile kann es sowohl Einnahmen als auch Ausgaben geben.

Spalte H: Projektnummer

Spalte I: Laufende Nummer

Die **Summenzeile** beinhaltet Formeln und „wandert“ nach unten. Die Tabelle selbst besitzt einen benannten Bereich („RechDB“), der alle Bewegungsdaten beinhaltet. Dieser wird beim Einfügen eines neuen Datensatzes per Makro entsprechend angepasst. Die Summenformel lautet dann (hier für die Einnahmen): `=DBSUMME(RechDB; "Einnahmen"; $Formeln.$F$3:$F$4)`

5.2.8 Tabelle Auswertungen

Die Auswertungstabelle bietet eine Übersicht aller Ein- und Ausgaben aufgliedert nach Monaten und Konten, sowie unterteilt nach Einnahmen und Ausgaben. Zusätzlich gibt es Spaltensummen.

Die Tabelle wird dynamisch verwaltet und zeigt somit immer die aktuellen Summen (gesteuert über Datenbank-Formeln), jedoch wird die Tabelle einmalig erstellt und die Konten sind dann statisch. Wurden neue Konten hinzugefügt oder entfernt, so muss die Tabelle neu aufgebaut werden. Hierzu steht jedoch ein Makro zur Verfügung.

Die Optionen der Tabelle verhindern im Normalfall die Anzeige der den Zellen hinterlegten Formeln. Um diese sichtbar zu machen, muss die Option entsprechend geändert werden.

Ein Beispiel für eine solche Formel lautet:

```
=WENN(DBSUMME(Jan; "Einnahmen"; $Formeln.$B$41:$B$42); DBSUMME(Jan; "Einnahmen"; $Formeln.$B$41:$B$42); "")
```

Diese Formel nutzt die Konto-Kriterienbereiche, wie sie in der Tabellen Formeln abgelegt wurden zur Filterung der Daten hinsichtlich des Kontos. Die Spalten (Monate) werden hingegen über die benannten Bereiche realisiert. Die Summenformeln hingegen sind einfach und werden über das Makro auf die tatsächlichen Verhältnisse angepasst.

Wichtig:

Die komplette Tabelle wird per Makro erzeugt. Manuelle Änderungen innerhalb der Tabellen werden spätestens bei der nächsten Aktualisierung überschrieben!

Auch diese Tabelle besitzt im Kopfteil drei Schaltflächen, um die wichtigsten Aufgaben zu erledigen:



Diese sind wie folgt verbunden:

Main.but_geheStart	Ergebnis.Ergebnisse_eintragen	Ergebnis.Auswertung_drucken
--------------------	-------------------------------	-----------------------------

Die Zeilen der Auswertungen basieren im Übrigen auf Formatvorlagen (so wie alle anderen Zeilen der Tabellen auch), diese können natürlich individuell angepasst werden.

Wichtig:

Die Namen der Formatvorlagen dürfen nicht geändert werden, die werden über Makros verwendet und über den Namen zugewiesen.

Die hier verwendeten Formatvorlagen sind:

Auswertung_Üb – grau hinterlegt, Strich oben und unten, für die Kopfzeile,

Auswertung_Üb_fett – wie oben, jedoch fette Schrift

Auswertung_Üb_fett_EUR – wie oben, jedoch zusätzlich als Währung formatiert

Auswertung_tab_re – Normale Zahlenformatierung in der Tabelle

5.2.9 Tabelle Projektkosten

Diese Tabelle dient dazu, einzelne Projekte in ihrer Übersicht darzustellen. Der Inhalt der Tabelle wird komplett von Makros erzeugt (bis auf die Kopfzeilen), insofern ist der aktuell angezeigte Stand immer mit Vorsicht zu genießen.



Die Hauptaufgaben lassen sich wiederum über Buttons am Kopf der tabelle erledigen. der erste Button – **Zurück zur Hauptseite** – ist wie gewohnt verbunden mit: [Main.but_geheStart](#)

Die anderen beiden Button sind verbunden:

Projekt wählen: [Projekte.Projektauswahl](#)

Auswertung drucken: [Projekte.projekt_drucken](#)

Die Felder **C2** (Projektnummer und Name) sowie **C4** (Projektklasse und Budget) werden jeweils per Makro nach der Wal des Projektes gefüllt, so wie auch der Rest der Tabelle.

Auch hier werden den jeweiligen Zeilen wieder diverse Formatvorlagen per Makro zugewiesen.

Die Projekttable wertet Projekte wie folgt aus:

Zunächst werden alle Belege aufgelistet (das Makro geht dabei durch die Monate Jan – Dez und wählt passende Datensätze aus), dann erfolgt eine Zwischensumme.

Anschließend werden die Rechnungen ausgewertet, es folgen Teilsummen, Gesamtsummen und ein Vergleich zum ursprünglichen Budget.

Belege:				
<u>Belegnr</u>	Datum	Beschreibung	Einnahmen	Ausgaben
2006-006	21.08.2006	Plan GmbH	150,00 €	
2006-007	22.08.2006	Rechnung	30,00 €	
2006-008	23.08.2006	Rechnung	10,00 €	
2006-009	24.08.2006	Test	250,00 €	
Summen Belege:			440,00 €	- €
Rechnungen:				
<u>Belegnr</u>	Datum	Beschreibung	Einnahmen	Ausgaben
Summen Rechnungen:			- €	- €
Summen Belege+Rechnungen:			440,00 €	- €
Gesamtsumme: 440,00 €			- Restbudget: 560,00 €	

Manuelle Änderungen in der Tabelle sind nicht zu empfehlen, da bei einer Neuwahl eines Projektes alles zunächst gelöscht wird und die Tabelle dann per Makro neu aufgebaut wird. Alle Änderungen gehen somit verloren.

5.2.10 Tabelle Kontenauswertung

Ähnlich wie die Projektauswertungen gibt es auch eine Kontenauswertung. Hierzu dient diese Tabelle.

	A	B	C	D	E
1					
2	Auswertung Konto:		5610 - Spenden	<input type="button" value="Konto wählen"/>	
3	<input type="button" value="← zurück zur Hauptseite"/>			<input type="button" value="Auswertung drucken"/>	
4			Einnahme-Konto - Budget: 1.000,00 €		
5	Belege:				
6	<u>Belegnr</u>	Datum	Beschreibung	Einnahmen	Ausgaben
7	2006-026	22.08.2006	Test	25,00 €	
8	2006-027	15.09.2006	Plan GmbH	86,00 €	
9	Summen Belege:			111,00 €	- €
10	Rechnungen:				
11	<u>Belegnr</u>	Datum	Beschreibung	Einnahmen	Ausgaben
12	Summen Rechnungen:			- €	- €
13	Summen Belege+Rechnungen:			111,00 €	- €
14	Gesamtsumme: 111,00 €			- Restbudget: 889,00 €	
15					
16					

Der Aufbau ist identisch, auch hier wird alles per Makros gesteuert. Manuelle Änderungen werden immer verworfen. Die Buttons sind wie folgt verbunden:

Zurück zur Hauptseite – : [Main.but_geheStart](#)

Konto wählen: [Konten.Kontoauswah](#)

Auswertung drucken: [Konten.Kontoauswertung_drucken](#)

Alles andere ist in den Makros selbst zu ändern oder anzupassen. Siehe hierzu auch die

Informationen zu den Modulen sowie im Code selbst.

5.2.11 Tabelle Kassenabgleich

Die letzte Tabelle erlaubt einen einfache Kassenabgleich. Diese Tabelle ist die einzige, die direkte Eingaben auch des Benutzers erwartet und akzeptiert. Alle gelb hinterlegten Felder können direkt editiert werden, alle anderen Felder werden durch Formeln berechnet.

Auch hier gibt es wieder diverse Schalter welche diverse Aktionen direkt auslösen:

<-- zurück zur Hauptseite		Abrechnung aktualisieren		Prüfbericht drucken	
Bitte zuerst Monat wählen:		August ▼		alle Felder leeren	

Die Buttons sind wieder mit Makros verbunden, und zwar:

Zurück zur Hauptseite – : [Main.but_geheStart](#)

Abrechnung aktualisieren: [Kasse.Kasse_aktualisieren](#)

Prüfbericht drucken: [Kasse.Kasse_drucken](#)

Auch das Auswahlfeld des Monats ist verbunden mit einem Makro – immer dann, wenn die Auswahl modifiziert wird:

Listenfeld für den Monat: [Kasse.Kasse_aktualisieren](#)

Der Button **alle Felder leeren** entfernt alle Eingaben der darunterliegenden Spalte, also der gezählten Geldbeträge. Verbunden ist er mit:

alle Felder leeren: [Kasse.Felder_leeren](#)

Auch für die anderen Eingabefelder gibt es bequeme Löschbuttons:

20	Einnahmen (noch nicht eingetragen):			1. Löschbutton verbunden mit: Kasse.EinFelder_Leeren gelöscht werden die Felder B22 bis C24
21	löschen	(Beschreibung)	(Betrag)	
22				
23				
24				
25	Ausgaben (noch nicht eingetragen):			2. Löschbutton verbunden mit: Kasse.AusFelder_leeren gelöscht werden die Felder B27 bis C31
26	löschen	(Beschreibung)	(Betrag)	
27	Vorschuss			
28	Vorschuss			
29	Vorschuss			
30	Vorschuss			
31	Vorschuss			
32				

Die Formeln der Tabelle sind einfach, in der Regel normale Summenformel. Lediglich die beiden Werte Einnahmen und Ausgaben werden per Makro berechnet und dann dort eingetragen (D15 und D16). Dies geschieht immer dann, wenn entweder der Button „Kasse aktualisieren“ gedrückt wird oder ein anderes Datum (Monat) gewählt wird.

5.3 Struktur der Makros

Alle Funktionalitäten sind in OpenOffice.org Basic realisiert, alle Prozeduren (Makros) sind gespeichert in der **Standard-Bibliothek** des Dokumentes. Innerhalb der Standardbibliothek werden verschiedene Module genutzt, hier zunächst die Übersicht:

Modulname	Beschreibung
_Lizenz	Dieses Modul ist nur informativ. Es enthält als Kommentar die Lizenzhinweise sowie die Versionskontrolle
Admin	Enthält Makros für den Administrator – speziell globale Makros
Belege_bearb	Makros zur Belegauswahl (Liste), zum Einlesen der vorhandenen Daten und zum ändern. Buttons der Auswahllisten
Daten	Datenmanipulation (Belege + Rechnungen), insbesondere das Speichern der Daten aus dem Belegdialog und die dazugehörigen Routinen
Ergebnis	Makros zum Erstellen der Ergebnistabelle (Auswertungen)
Helper	Hilfsfunktionen
Init_Mod	Initialisierungsmodul. Hier werden die globalen Variablen deklariert und die Hauptinitialisierung durchgeführt.
Kasse	Makros für den Kassensabgleich
Konten	Makros speziell für die Konten – Verwaltung und Auswertung
Main	Hauptmodul mit den Makros für den Beleg-Dialog
Projekte	Makros speziell für die Projekte – Verwaltung und Auswertung
Rechnungen	Makros speziell für Rechnungen – Verwaltung und Auswertung

Darüber hinaus beinhaltet die Bibliothek diverse Dialogmodule:

Modulname	Beschreibung
Beleg	Hauptdialog. Belege erfassen, ändern, Rechnungen erfassen und ändern. Beleg umfasst vier sichtbare Stufen (Seiten) und eine unsichtbare.
DlgListe	Auswahl-Liste, wird per Makro auf die unterschiedlichen Auswahlen vorbereitet.
Initdlg	Stammdaten-Dialog
KtoDlg	Dialog „Neues Konto anlegen“, auch zum ändern
ProjDlg	Dialog Projekte – zum Neuanlage oder auch zum Ändern.
PWdlg	Passwort-Dialog
Statusbar	Statusbalken – Rückmeldung bei längeren Makros
Wait	Info Dialog für Wartezeiten

Nach dieser ersten Übersicht nun die Details der jeweiligen Module ausgenommen des Moduls _Lizenz, der keine Funktionalität enthält. Es empfiehlt sich aber, den Inhalt einmal zu lesen:

5.3.1 Admin

Funktion	Kurzbeschreibung
Admin	nur für direkten Aufruf der Makros aus der IDE – Startmakro
But_del_all	Button „Löscht alle Datensätze“ - löscht die Bewegungsdaten
Bereiche_definieren	Legt die benannten Datenbereiche über alle Monatstabellen und die Rechnungen an. Bestehende werden zunächst gelöscht.
Rechnungen_loeschen	löscht alle Rechnungen
Admin_modus	Schaltet in den Administratormodus oder diesen aus.
Sperren_aufheben	Setzt die Belegsperrern zurück
NeuBerechnenInterneFormeln	hierdurch werden alle internen Formeln neu berechnet, ruft insgesamt vier Einzelmakros auf, die das denn jeweils erledigen.

5.3.2 Belege_bearb

Funktion	Kurzbeschreibung
But_BelegeBearbeiten	Button „Belege bearbeiten“ des Belegdialoges. Startet die entsprechenden Makros.
But_DatVerw	Button „Belege_verwalten“ der Monatsjournal-Tabellen.
Auswahldialog()	ruft den Auswahldialog auf und konfiguriert diesen zunächst entsprechend. Erwartet 5 Parameter.
mittlerer_Button	mittlerer Funktionsbutton des Auswahldialogs – ruft die entsprechenden Funktionen auf.
rechter_Button	rechter Funktionsbutton des Auswahldialogs – ruft die entsprechenden Funktionen auf.
rechterButton_Belege	rechter Funktionsbutton bei Belegauswahl-Dialog
Journal_bearbeiten	Monatsbelege bearbeiten – Aufruf über Buttons der Monatstabellen
Journal_liste_erzeugen	Monatsbelegliste erzeugen – zur Darstellung im Auswahldialog.
Beleg_uebernehmen()	Der gewählte Beleg wird in die Belegbearbeitungsmaske eingetragen (erwartet 2 Parameter)
Monatswechsel	Monatswechsel bei Belegauswahl-Dialog
Beleg_stornieren	aktueller Beleg wird storniert (Betrag = 0 gesetzt) – Button in

	Belegdialog
Beleg_stornieren2	aktueller Beleg wird storniert (Betrag = 0 gesetzt) – Button in Auswahldialog.

5.3.3 Daten

Funktion	Kurzbeschreibung
Werte_speichern	Button „Werte speichern und weiter“ der Belegmaske – ruft Werte_speichern2 auf
Werte_speichern_ende	Button „Werte speichern und Schliessen“ der Belegmaske – ruft Werte_speichern2 auf
Werte_speichern2()	Routine zum Speichern der Daten in der Belegmaske
add_Datensatz()	Ein Datensatz wird hinzugefügt
findeLetztenEintrag()	extrahiert die letzte Eintragszeile der Tabelle (wird als Parameter übergeben) und liefert den entsprechenden Zeilenindex
change_db()	definiert den benannten Bereich einer Tabelle neu, erwartet drei Parameter.
Fix_neue_Nummer()	Die Belegnummer eines neuen Beleges wird in den Optionen gespeichert.

5.3.4 Ergebnis

Funktion	Kurzbeschreibung
Ergebnisse_eintragen	Diese Funktion schreibt die komplette Auswertung der Tabellen Auswertungen – alle Formeln und Zeilen. Ruft die folgenden Funktionen mit auf.
ZeileSchreiben()	Einen einzelnen Zeile wird in der Auswertung geschrieben – erwartet vier Parameter
ZeileReSchreiben()	Die Zeile für die Rechnungen wird geschrieben – erwartet drei Parameter
Auswertung_drucken	Bereitet die Druckausgabe der Auswertung vor und druckt diese.

5.3.5 Helper

Funktion	Kurzbeschreibung
aktuellerPfad()	extrahiert den aktuellen Pfad des Dokumentes – welches als Objekt übergeben wird
NumAdd()	Erhöht eine alphanumerische Zeichenfolge im letzten

	Ziffernblock um 1 (hochzählen der Beleg- und Rechnungsnummern) – erwartet zwei Parameter
autoCalcOff	automatische Berechnung abschalten
autoCalcOn	automatische Berechnung einschalten
Protect_alt	Zellschutz wird aktiviert – aktuell nicht genutzt
Protect_Tab()	Zellschutz wird aktiviert/deaktiviert – erwartet zwei Parameter
getLastRow()	Liefert die letzte benutzte Zeile eines Zellbereichs (Index)
checkProjekt	prüft, ob die Projektliste im Belegdialog aktiviert wird oder nicht
Button_heute	schreibt das aktuelle Datum(System) in den Belegdialog.
Button_reset	Setzt alle Felder des Belegdialoges zurück
JournalZellenFormatieren()	Spaltenformatierung der Journalzeilen – erwartet zwei Parameter
set_Uebertrag()	Schreibt die Übertragsformeln der Monatsjournale neu – erwartet zwei Parameter
Journal_drucken	Bereitet die Druckausgabe des Monatsjournals vor und druckt dieses aus.
umsortieren()	sortiert eine Liste (Array) um – der letzte Eintrag wird der erste und so weiter. Erwartet die Liste als Parameter.
EntferneFuehrendeZiffern()	entfernt aus einem übergebenen String führende Leerzeichen und Ziffern
PW_Kontrolle	Passwortkontrolle
bruttofeld	Setzt die Bruttofeld-Anzeige im Belegdialog
But_Neue_Nummer	fordert eine neue Nummer für den angezeigten Beleg an – dieser wird dadurch auch ein neuer Beleg.
noshow	Hinweis bei noch nicht realisierten Funktionen
speichern	Dokument wird gespeichert – falls es noch keine URL hat, Speicherdialog aufrufen.
Kopfzeilen_anpassen	Kopfzeilen der Berichte füllen
Dat_Check	Datumsüberprüfung bei zu ändernden Belegen
EingabeFelderSperrern()	Felder der Belegmaske sperren / entsperren
Statusdialog()	Fortschrittsbalken Dialog erzeugen und anzeigen
Status_akt()	Fortschrittsbalken im Dialog aktualisieren

5.3.6 Init_Mod

In diesem Modul werden alle globalen Variablen definiert.

Diese sind alle mit einem Kommentar versehen und in Blöcke aufgeteilt:

Hier jetzt alle in der Übersicht:

```
REM Monatslisten
Global Const cMonStartRow = 3 'Erste Datenzeile der Monatslisten (Kopfzeile)
```

```
'-Achtung: Index!
Global Const cMonEndBlockRows = 5 'Zeilenanzahl des Schlussblocks
Global Const cMonStartColumn = 0 'Erste Spalte des Datenbereiches (Index)
Global Const cMonEndColumn = 8 'Letzte Spalte des Datenbereiches(Index)
Global Const cSaldoVorAdrE = "$F$1" 'Zelladresse des Saldovortrages Einnahmen
Global Const cSaldoVorAdrA = "$G$1" 'Zelladresse des Saldovortrages Ausgaben
Global Const cDruckLfdNr = 0 'Laufende Nummer wird gedruckt (1) od. nicht (0)

REM Konten
Global Const cKtoStartRow = 7 'Erste Zeile der Kontoliste (Absolut)
Global Const cKtoArtColumn = "B" 'Spalte der Kontenart
Global Const cKtoNrColumn = "C" 'Spalte der Kontonummer
Global Const cKtoColumn = "D" 'Spalte der Kontobeschreibung
Global Const cKtoFormel = "G" 'Spalte der Bereichs-Formeln
Global Const cKtoAktivColumn = "E" 'Spalte der Konto Aktiv Bezeichnung
Global Const cKtoAuswColumn = "F" 'Spalte der Konto Auswertung
Global Const cAnzKtoNr = 4 ' Stellenanzahl der Kontonummern/ Projektnummern

rem Beschreibungen/Firmen
Global Const cBesStartRow = 4 ' Erste Zeile der Beschreibungs - Firmenliste
Global Const cBesFixRows = 6 ' Fixe Einträge (werden nicht dynamisch
geändert
Global Const cBesColumn = "H" ' Spalte der Beschreibungen/Firmen
Global Const cBesEndRow = 29 ' Letzte Zeile, diese minus erste Zeile minus
' Fixzeilen, bestimmen den dynamischen Part - Rollierend

REM Projekte
Global Const cPrjStartRow = 7 ' Erste Zeile der Projektliste
Global Const cPrjNrColumn = "A" 'Spalte der Projektnummer
Global Const cPrjBesColumn = "B" 'Spalte der Projektbeschreibung
Global Const cPrjBudColumn = "C" 'Spalte des Projektbudgets
Global Const cPrjEinColumn = "D" 'Spalte der Projekt-Einnahmen
Global Const cPrjAusColumn = "E" 'Spalte der Projekt-Ausgaben
Global Const cPrjProzColumn = "F" 'Spalte der Projektbudget Erfüllung in %
Global Const cPrjRestColumn = "G" 'Spalte des Rest-Projektbudgets
Global Const cPrjAktivColumn = "I" 'Spalte der Bezeichnung Projekt aktiv
Global Const cPrjTypColumn = "H" 'Spalte des Projekttyps

REM Optionen
Global Const cBelegAdr = "L8" 'Zelladresse der Belegnummer in Tab Stammdaten
Global Const cRechAdr = "L11" 'Zelladresse der Rechnungsnummer in Stammdaten
Global Const cPrjAdr = "L14" 'Zelladresse der Projektnummer in Tab Stammdaten
Global Const cLfdnrAdr = "L2" 'Zelladresse der laufenden Nummer in Stammdaten
Global Const cDatumAdr = "L5" 'Zelladresse des letzten Eingabedatums in Stammd
```

```

Global Const cLogoLAdr = "L19"      'Zelladresse des Dateinamens linkes Logo
Global Const cLogoRAdr = "L20"      'Zelladresse des Dateinamens rechtes Logo
Global Const cNameAdr = "C7"        'Zelladresse des Namens der Einrichtung
Global Const cKStAdr = "C8"         'Zelladresse der Kostenstelle / Name
REM Beschriftungen
Global Const cTxtBeleg1 = "Belege des Monats..."
Global Const cTxtBeleg2 = "KtoNr  KtoBez          Datum      BelegNr
Beschreibung      Betrag    E/A  ProjektNr"
Global Const cTxtButBeleg1 = "Beleg bearbeiten"
Global Const cTxtButBeleg3 = "Beleg stornieren"

'Globale Variable definieren
global oDoc as Object                'das Calc-Dokument
global oOptTab as Object             'das Calc-Tabelle Stammdaten
global oKtoTab as Object             'die Calc-Tabelle Konten
global oPrjTab as Object             'die Calc-Tabelle Projekte
global oForTab as Object             'die Calc-Tabelle Formeln (ausgeblendet)
global oReTab as Object             'die Calc-Tabelle Rechnungen
global oErgTab as Object            ' die Tabelle Auswertungen
global oPrjAusTab as Object          'die Tabelle Projektauswertung
global oKtoAusTab as Object          'die Tabelle Kontoauswertung
global oKasseTab as Object           'die Tabelle Kassenbuch-Abgleich
global sPW as string                'Admin-Passwort
global oDlg as object               'der Beleg-Dialog
global oIDlg as object              'der Init-Dialog
global oPDlg as object              'der Projekt-Dialog
global oKDlg as object              'der Konto-Dialog
global oPWDlg as Object             'der Passwort-Dialog
global oStDlg as Object             'der Fortschritts-Dialog
global aMonate()                   'Liste der Monatsnamen
global aMon()                       'Liste der verkürzten Monatsnamen
global bInitFlag as boolean         'Initalisierungsflag
global bStorno as boolean           'Stornierungsflag
global bAdmin as boolean            'Admin-Flag
global bSperr as boolean            'Flag, wenn Belegsperr aktiv
global bBelegChange as boolean      'Flag - für Änderungen eines Datensatzes
global iKenn as integer             'Kennziffer für Belegwahl 1 = Belege in
                                     ' Maske, 2 = Belege ohne Maske
                                     '3 - Rechnungen in Maske, 4 - Belege ohne
                                     ' Maske, 5- Projekte, 6 - Konten

```

Weiterhin finden sich im Modul die folgenden Makros:

Funktion	Kurzbeschreibung
init	Initialisierung des Programms und der Variablen
prog_start	Funktion ist mit dem Dokumentenstart verbunden
Stammdaten	Aufruf des Stammdaten-Dialogs
Start_InitDlg()	Der Stammdaten-Dialog
admin_mode_but	Schalter „Administrator_Modus“ im Stammdatendialog
change_AdminModus()	Wechselt in oder aus dem Administrator-Modus und nimmt die eintsprechenden Einstellungen vor.

5.3.7 Kasse

Funktion	Kurzbeschreibung
Kasse_aktualisieren	aktualisiert die im Kassenabgleich dargestellten Summen für Einnahmen und Ausgaben entsprechend dem Zeitraum
Felder_leeren	Löscht die Inhalte der Eingabefelder im Kassenabgleich – hier Anzahl der gezählten Geldstücke und Scheine
EinFelder_leeren	Löscht die Inhalte der Eingabefelder im Kassenabgleich – hier nicht gebuchte Einnahme-Felder
AusFelder_leeren	Löscht die Inhalte der Eingabefelder im Kassenabgleich – hier nicht gebuchte Ausgabe-Felder
Kasse_drucken	Kassenprüfbericht drucken

5.3.8 Konten

Funktion	Kurzbeschreibung
KontenBereicheSchreiben	erneuert bzw. schreibt die Kriterienbereiche für alle Konten und trägt die Bezugsgrößen bei den Konten ein
Kontoauswahl	Button „Konto wählen“ aus der Tabelle – ruft den Auswahldialog auf.
Kontoliste_erzeugen	Erzeugt die Kontoliste für den Auswahldialog
Konten_Auswertung	Erstellt die Kontoauswertung
Kontoauswertung_drucken	Druckt die Kontoauswertung
KontoSaldo_akt()	aktualisiert die Kontensalden bei Eingabe/Änderung von Belegen
Konten_verwalten	Startet Kontobearbeitungsdialog nach Auswahl des Kontos
KontoHinzufuegen	Startet Kontobearbeitungsdialog aus den Stammdaten
KtoDaten_eintragen()	geänderte oder neue Kontendaten werden in die Kontoliste eingetragen

AdminKtoDlg	Administratormodus aus Kontobearbeitungsdialog
-------------	------------------------------------------------

5.3.9 Main

Funktion	Kurzbeschreibung
But_Belege_eingeben	Button „Belege eingeben“ aus Tabelle Stammdaten
But_Rechnungen_eingeben	Button „Rechnungen eingeben“ aus Tabelle Stammdaten
Belegdialog()	Startet den Belegdialog
initBelegDialog	initialisiert den Belegdialog und füllt die entsprechenden Daten und Listfelder.
fillBeschreibung	füllt die Liste der Beschreibungen
onChangeEinAus	wird ausgeführt, wenn bei vorhandenen Daten im Belegdialog die Art des Beleges (Einnahme/Ausgabe) geändert wird.
wechselKonto	wird ausgeführt, wenn das Konto gewechselt wird
WechselFirma	wird ausgeführt, wenn die Beschreibung/Firma geändert wird
fillProjekt	Projektliste wird gefüllt
Projekt_pruefen	prüft, ob das eingegebene Projekt schon existiert oder nicht
button_text	die Multifunktionsschalter erhalten ihre Beschriftung
but_erfassen	Dialog_Button Belege/Rechnungen erfassen
but_zurueck	Dialog Button zurück
onBetragChange	Betrag ändern -Ereignis Text modifiziert
but_geheAuswertung	Schalter „Gehe zu Auswertung“ im Stammdaten
but_gehePrjAuswertung	Schalter „Gehe zu Projektauswertung“ im Stammdaten
but_geheKtoAuswertung	Schalter „Gehe zu Kontoauswertung“ im Stammdaten
but_geheKasseAuswertung	Schalter „Gehe zu Kassenauswertung“ im Stammdaten
but:geheStart	Schalter „Gehe zu Startseite“ in den Tabellen

5.3.10 Projekte

Funktion	Kurzbeschreibung
Projektdaten_akt()	die Projektsalden Liste wird angepasst bzw. neu geschrieben
np_aus_Dialog	Neues Projekt aus Dialogmaske – ruft „NeuesProjekt“
np_ausTabelle	Neues Projekt aus Tabelle – ruft „NeuesProjekt“
NeuesProjekt	Startet Dialogmaske „Neues Projekt“
Projektdaten_eintragen	Projektdaten aus dem Dialog werden in die Liste eingetragen
Projekt_liste_erzeugen	Projektliste für Auswahldialog erstellen
Projektkosten_Liste	Projektauswertung erzeugen
but_autProjNummer	Nächste verfügbare Projektnummer wird eingetragen

projekt_drucken	Projektauswertung drucken
ProjektSaldenNeuschreiben	Aktualisieren/Neuschreiben allerProjektsalden
Projekt_verwalten	läd ein ausgewähltes Projekt in den Projektdialog – zum Bearbeiten

5.3.11 Rechnungen

Funktion	Kurzbeschreibung
But_rechnungsverwaltung	Button Rechnungsverwaltung – ruft das Rechnungsjournal auf
ReJournal_bearbeiten	Die Rechnungsliste für den Auswahldialog wird erzeugt
mittlerer_Button_Re	mittlerer Funktionsbutton des Auswahldialogs
rechterButton_Rechnungen	rechter Funktionsbutton des Auswahldialoges

5.4 Die Dialoge

Neben den Programm-Modulen gibt es auch noch die Dialog-Module. Diese werden im folgenden vorgestellt, wobei sie ergänzt werden durch Informationen über den Namen der Kontrollelemente, die Art des Kontrollelementes sowie über die verknüpften Makros. Dort, wo es unterschiedliche Seiten (Steps) gibt, wird dieses ebenfalls angezeigt.

Die Felder erhalten also eine Kurzbezeichnung in der Art: Name (des Kontrollelementes), Typ, Seite (Step). Steht dort also so etwas wie **belegnr, Text, 0** dann bedeutet das, dass der Name des Kontrollelementes (und somit des Objektes) „belegnr“ lautet, es sich um ein Textfeld handelt und dieses auf der Ebene 0 platziert ist (die Ebene 0 ist immer sichtbar).

Zusätzlich werden verbundene Aktionen in der Form **Modul.Makro** angezeigt.

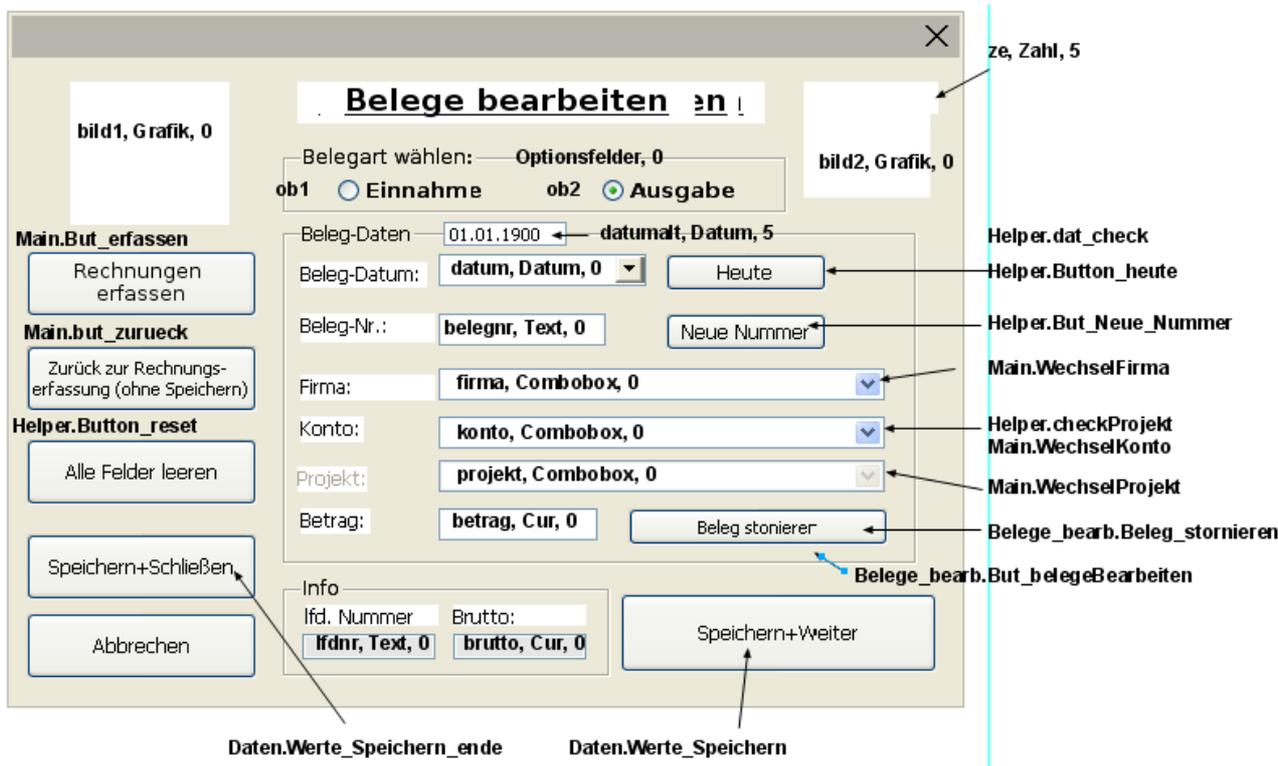
5.4.1 Beleg

Der wichtigste Dialog ist die Belegerfassung. Dieser Dialog wird sowohl für die Belegerfassung (Barbelege) als auch für die Belegänderung genutzt, aber auch für die Rechnungserfassung und -änderung. Insofern besteht der Dialog aus vier sichtbaren Seiten (Step) deren Felder ähnlich, aber nicht gleich sind. Weiterhin beinhaltet der Dialog einige nicht sichtbare Steuerelemente, die dann auf Seite 5 (Step 5) angeordnet sind.

Die Überschrift des Dialoges ist jedesmal ein Beschriftungsfeld (Label), wobei der Inhalt pro Seite unterschiedlich ist. Durch den Inhalt lässt sich auch gleich die Verwendung des Dialoges passend zu der Step-Eigenschaft feststellen:

Ebene (Step) des Dialoges	Überschrift (Label)	Bemerkung
Step 0	-	Grundseite, trägt die Buttons und die überall sichtbaren Elemente
Step 1	Belegerfassung	Dialog für neue Barbelege
Step 2	Belege bearbeiten	Dialog für das Verändern bestehender Belege

Step 3	Rechnungen erfassen	Dialog zum Erfassen neuer Rechnungen
Step 4	Rechnungen bearbeiten	Dialog zum Verändern bereits gebuchter Rechnungen
Step 5	-	trägt die versteckten Kontrollelemente



Die beiden oberen Buttons auf der linken Seite erhalten ihre Beschriftung entsprechend der Step-Eigenschaft per Makro. Zuständig hierfür ist das Makro **Main.button_text** in dem auch bei Bedarf die Texte angepasst werden können. Insofern besitzen auch die beiden Buttons eigene Namen:

	stepbut
	bebut

Der **Abbrechen** Button ist mit keiner Aktion verbunden, sondern als Typ „Abbrechen“ definiert. Dadurch wird der Dialog beendet und das aufrufende Makro fortgesetzt.

5.4.2 DlgListe

Ein weiterer multifunktionaler Dialog ist die Auswahlliste, ein einfacher Dialog zum Auswählen von

Datensätzen. Diese wird sowohl für Belege und Rechnungen, aber auch für Konten und Projekte verwendet. Der Dialog ist so aufgebaut, dass in der Regel alle Elemente erst durch Makros ihre angezeigten Eigenschaften (Titel etc) erhalten.



Diese Dialog besitzt einige Eigenheiten: Das Textfeld „typ“ wird nur intern genutzt und dient als Zwischenspeicher der gerade genutzten Typs diese Dialoges. Diese Feld ist nie sichtbar, wird aber per Makro ausgeschaltet. Dabei gibt es folgende „Typen“:

Typ	Bedeutung
1	Belegauswahl, Monatslistbox ist sichtbar und aktiv
2	Rechnungsauswahl
3	Projektauswahl
4	Kontenauswahl

Die Makros der unteren Buttons verzweigen je nach Typ auf unterschiedliche Aktionen.

Der Dialog **DlgListe** wird durch die Funktion **AuswahlDialog()** im Modul **Belege_bearb** aufgerufen und aufgebaut. Die Funktion erwartet dabei 6 oder 7 Übergabeparameter, die dann auch das Aussehen des Dialoges steuern:

Parameter	Typ	Bedeutung
1	String	Überschrift – Inhalt des Feldes text1
2	String	Listenüberschrift – Inhalt des Feldes text2
3	String	Schalterbeschriftung rechts – Label des Objektes but1

4	String	Schalterbeschriftung mite – Label des Objektes but3
5	String	Typ – Inhalt des Feldes typ
6	Array	Liste der Einträge – Inhalt des Feldes liste
7 (optional)	String	Monat – Vorgabe für die Listbox Monate

Der komplette Code hier:

```

sub Auswahldialog(slb1 as string, slb2 as string, sCB1 as string, _
                sCB2 as string, styp as string, aListe, _
                optional sMonat as String)

    oBDlg = CreateUnoDialog(DialogLibraries.Standard.DlgListe)
    oBDlg.getControl("text1").text = slb1
    oBDlg.getControl("text2").text = slb2
    oBDlg.getControl("but3").model.Label = sCB1
    oBDlg.getControl("but1").model.Label = sCB2
    oBDlg.getControl("typ").text = styp
    oBDlg.getControl("typ").Visible = false
    oBDlg.getControl("liste").model.stringItemList = aListe()
    With oBDlg.getControl("monate")
        .model.stringItemList = aMonate()
        .removeItems(0 , 1)
    end with
    if not (sTyp = "1") then oBDlg.getControl("monate").visible = false
    if not isMissing(sMonat) then
        oBDlg.getControl("monate").selectItem(sMonat, true)
    end if
    oBDlg.execute()
End sub

```

5.4.3 Initdlg

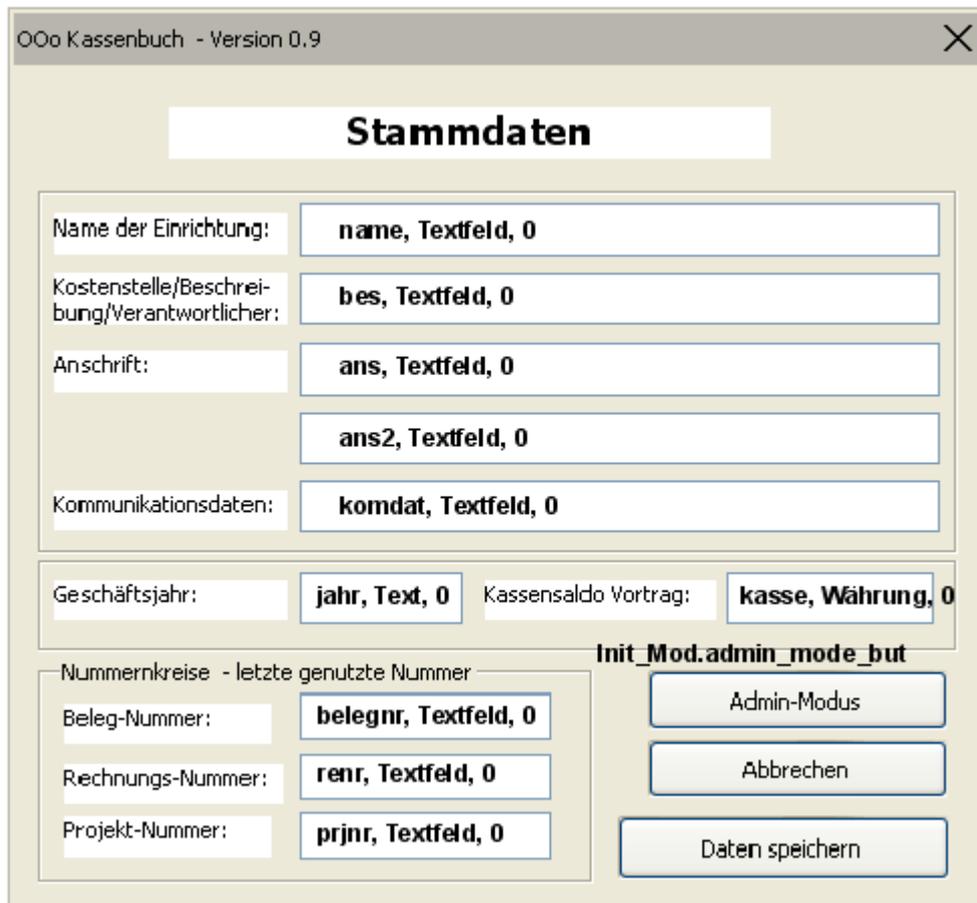
Der **Initdlg** wird beim ersten Start des Programms automatisch aufgerufen und dient zur Aufnahme der Stammdaten. Werden diese später eingegeben oder geändert, wird dieser Dialog ebenfalls genutzt.

Die Schaltflächen „Abbrechen“ und „Daten speichern“ sind nicht mit direkten Makro-Aktionen verbunden, sondern als spezielle Schaltflächen eingerichtet:

Abbrechen - als Schaltfläche Abbrechen

Daten speichern als Schaltfläche OK

Dadurch liefern Sie unterschiedliche Rückgabewerte und diese werden dann per Makro ausgewertet und entsprechend darauf reagiert.



OOo Kassenbuch - Version 0.9

Stammdaten

Name der Einrichtung:

Kostenstelle/Beschreibung/Verantwortlicher:

Anschrift:

Kommunikationsdaten:

Geschäftsjahr: Kassensaldo Vortrag:

Nummernkreise - letzte genutzte Nummer

Beleg-Nummer:

Rechnungs-Nummer:

Projekt-Nummer:

Init_Mod.admin_mode_but

Der Dialog wird durch die Funktion **Start_initDlg()** im Modul **Init_Mod** aufgerufen. Ist es nicht die Erstinitialisierung, dann werden die Kontrollelemente durch die Inhalte der Stammdaten und der Optionen gefüllt.

5.4.4 KtoDlg

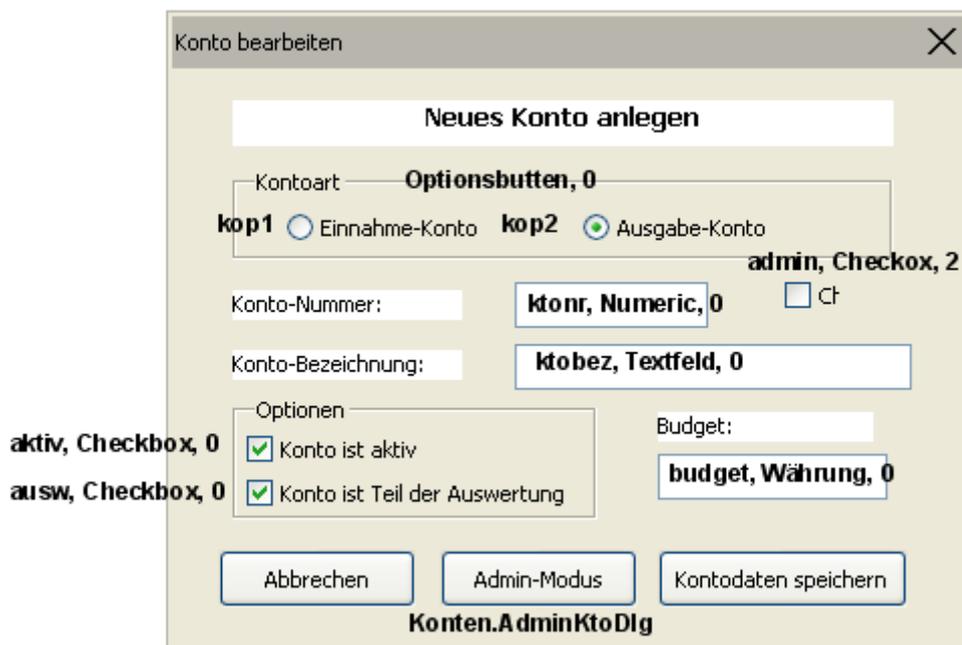
Über den Dialog **KtoDlg** werden die Konten angelegt oder verwaltet. Viele der Felder sind nur aktiv, wenn der Administrator-Modus eingeschaltet wird.

Auch dieser Dialog arbeitet mit unterschiedlichen Seiten (Ebenen) um Steuerfelder zu nutzen, die aber nicht zu sehen sein sollen.

Insgesamt gibt es 2 Ebenen (Ebene 1 und Ebene 2), die Ebene 1 ist die normale Arbeitsebene, in Ebene 2 befindet sich nur ein Steuerelement, **admin**, die Checkbox.

Dieses Steuerelement wird nur dazu genutzt, festzuhalten, dass der Admin-Modus innerhalb dieses Dialoges eingeschaltet wurde. Dadurch besteht beim Schliessen des Dialoges, die Möglichkeit, den Admin-Modus auch direkt wieder auszuschalten. Geregelt wird dies mit einer Abfrage des Status dieser Checkbox und einer entsprechenden Sicherheitsanfrage.

Details hierzu siehe auch in der Funktion **KtoDaten_eintragen()** im Modul **Konten**.



Konto bearbeiten

Neues Konto anlegen

Kontoart — Optionsbutten, 0

kop1 Einnahme-Konto **kop2** Ausgabe-Konto

admin, Checkox, 2 **ct**

Konto-Nummer: **ktonr, Numeric, 0**

Konto-Bezeichnung: **ktobez, Textfeld, 0**

Options

aktiv, Checkbox, 0 Konto ist aktiv

ausw, Checkbox, 0 Konto ist Teil der Auswertung

Budget: **budget, Währung, 0**

Abbrechen Admin-Modus Kontodaten speichern

Konten.AdminKtoDlg

Der Button **Abbrechen** ist wieder auf Modus „Abbrechen“ gestellt, der Button **Kontodaten speichern** auf Modus „OK“.

Der Dialog wird über zwei unterschiedliche Funktionen gestartet, beide untergebracht im Modul **Konten**:

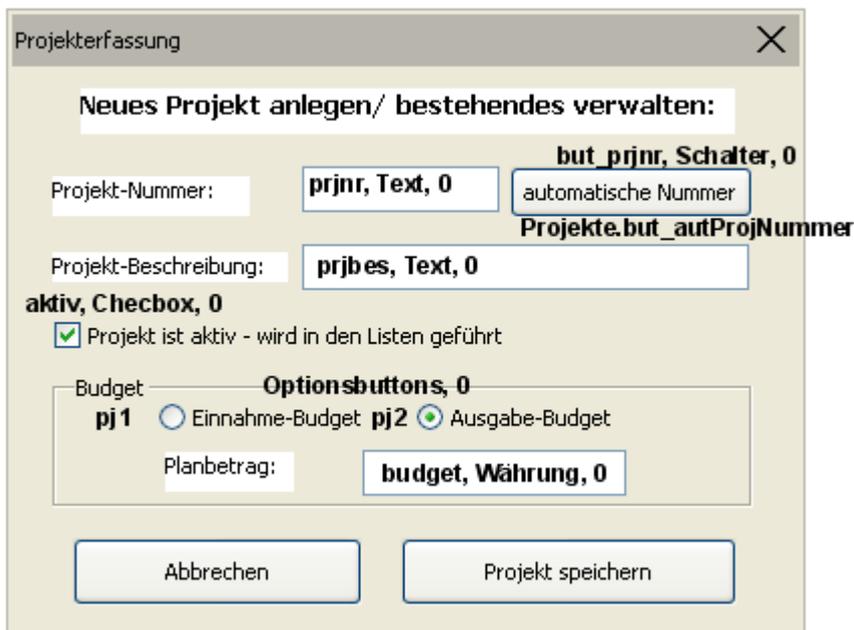
Konten_verwalten – ruft den Dialog mit den Daten des gerade ausgewählten Kontos auf.

KontoHinzufügen – ruft den Dialog „nackt“ auf zum Hinzufügen von neuen Konten.

Über den **Admin-Modus** Button wird der Passwort-Dialog aufgerufen, der den Wechsel in den Administrator-Modus ermöglicht. Gleichzeitig wird die Checkbox **Admin** auf State 1 gesetzt (angehakt), so dass nach Schliessen des Kontodialoges die Rückfrage erfolgen kann, ob der Admin-Modus auch wieder ausgeschaltet werden soll.

5.4.5 ProjDlg

Der Projektdialog **ProjDlg** dient ähnlich wie der Kontodialog zum Hinzufügen oder zum Ändern von Projekten.



Hinweis:
 Der Button automatische Nummer ist in einigen Versionen nicht aktiv!

Auch hier sind die unteren beiden Buttons nicht mit direkten Routinen verknüpft, sondern regeln über Standardrückgabewerte den weiteren Verlauf.

Der **Abbrechen** – Button ist als Schaltfläche „abbrechen“ konfiguriert, der **Projekt speichern** Button als Schaltfläche „OK“.

5.4.6 PWdlg

Der Passwort-Dialog PWdlg übernimmt die Überprüfung des Passwortes. Er stellt eine Eingabemöglichkeit für das Passwort dar, wobei die dargestellten Zeichen maskiert werden - mit dem Stern („*“) - und somit der eingegebene Text nicht sichtbar ist.



Es gibt nur diese eine Eingabefeld, die beiden Buttons sind wieder so konfiguriert, wie auch die Bezeichnung es anzeigt. Dadurch liefern sie wieder unterschiedliche Rückgabewerte, die dann die Makrosteuerung auswertet.

Der Dialog wird aufgerufen über die Funktion `admin_mode_but` im Modul `Init_mod`, wobei diese Funktion selbst entweder über

den Button „Admin-Mode“ des Stammdaten-Dialoges aufgerufen wird oder indirekt über das Makro `Admin_modus` aus dem Modul `Admin`, die wiederum verknüpft ist mit dem entsprechenden Schalter auf der Startseite im Verwaltungsmenü.

Ein Blick auf die Funktion `admin_mode_but`:

```

REM Button Admin - aus Init Dialog
sub admin_mode_but
    dim n%
    oPWdlg = CreateUnoDialog(DialogLibraries.Standard.PWdlg)
    n = oPWdlg.execute()
    if n = 0 then 'Abbruch
        exit sub
    end if
    REM Passwortkontrolle
    if not PW_Kontrolle then ' PW stimmt nicht!
        msgbox ("Das Passwort ist nicht richtig!" , 16, "Fehler!")
        exit sub
    end if
    if not isNull(oIdlg) then oIdlg.endexecute()
    change_AdminModus(true) 'Adminmodus einschalten
end sub

```

Die Passwort-Kontrolle selbst übernimmt die Funktion `PW_Kontrolle` aus dem Modul `Helper`:

```

REM Passwortkontrolle
function PW_Kontrolle as Boolean
    PW_Kontrolle = false
    if oPWdlg.getControl("pw").text = sPW then PW_Kontrolle = true
end function

```

Diese übernimmt einen einfachen Vergleich mit dem in der Zelle **B10** der Tabelle Formeln gespeicherten Wert. Dieser Wert wird während der Initialisierung (Funktion `init` im Modul `Init_mod`) ausgelesen und in der Variablen `sPW` gespeichert.

Auszug aus der Funktion `init`:

```

...
    sPw = oForTab.getCellRangeByName("B10").string
...

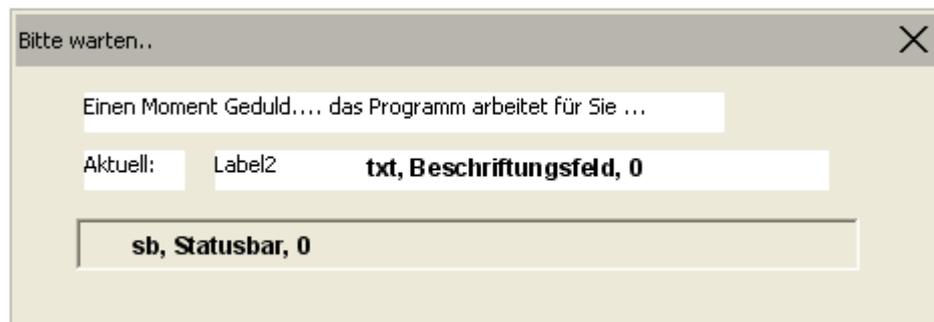
```

5.4.7 Statusbar und Wait

Bleiben noch zwei selbstdefinierte Dialoge, die aber keine Eingabemöglichkeit mehr haben sondern lediglich zu Informationszwecken genutzt werden.

Der Fortschrittsbalken (Statusbar) erhält seine Informationen per Makro – und auch den Stand es Fortschritts. Die entsprechenden Kontrollelemente werden insofern so angesteuert.

Ein Beispiel zeigt exemplarisch die Verwendung, wobei die Programmteile teilweise ausgeblendet werden, die nicht Bestandteil der Ansteuerung sind.



Geändert wird immer der Inhalt des Beschriftungsfeld **txt** sowie die Position des Statusbalkens. Das Beispiel zeigt den Einsatz des Statusbalkens beim Neuschreiben der benannten Bereiche. Dabei wird über eine Schleife jede Monatstabelle (12 Stück) sowie die Rechnungstabelle (+ 1) durchlaufen. Bei jedem Schleifendurchlauf soll der Statusbalken ein Segment mehr anzeigen – bei insgesamt 14 Segmenten:

(Funktion `Bereiche_definieren` im Modul `Admin`):

```

sub Bereiche_definieren
    dim iZe as long, oTab as object, sBereich as string
    Dim i%
    if not binitflag then init
    StatusDialog(1, 13)                'Fortschrittsbalken starten
    thisComponent.lockControllers()
    'Datenbereiche anlegen
    for i = 1 to uBound(aMonate)
        Status_akt("Bearbeite Monat " & aMonate(i), 1) 'FB aktualisieren
        sBereich = aMon(i)
        oTab = thisComponent.sheets.getByname(aMonate(i))
        iZe = findeletztenEintrag(oTab)+1 'Modul Daten
        change_db(oTab, sBereich, iZe) 'Modul Daten -
        Set_Uebertrag(iZe+cMonEndBlockRows, i) 'Übertragsformeln anpassen
    next
    'Datenbankbereich Rechnungen anlegen
    Status_akt("Bearbeite Rechnungen", 1) 'Fortschrittsbalken aktualisieren
    iZe = findeletztenEintrag(oReTab)+1
    change_db(oReTab, "RechDB", iZe) 'Modul Daten -
    thisComponent.UnlockControllers()
    oStdlg.dispose()                'Fortschrittsbalken Dialog schliessen
end sub

```

Anders als die bisherigen Dialoge darf der Start des Fortschrittsbalkens keinen Stop der Makroausführung bewirken. Der Start und die Initialisierung erfolgt in einer eigenen Funktion (`StatusDialog`) im Modul `Helper`. Diese Funktion erwartet zwei Parameter: den Startpunkt des Balkens und den maximalen Punkt des Balkens:

```

Sub Statusdialog(iStepStart as long, iStepEnd as long)
    if not binitflag then init
    oStdlg = CreateUnoDialog(DialogLibraries.Standard.StatusBar)
    oStdlg.getControl("sb").setRange(iStepStart, iStepEnd)
    oStdlg.setVisible(true)
end sub

```

Mit `setVisible(true)` wird der Dialog im Vordergrund angezeigt, die Makros aber laufen ohne Unterbrechung weiter. Der Dialog muss dann also manuell (per Makro) geschlossen werden – zum Beispiel über die Methode `dispose()`, wobei dadurch auch gleich das Objekt aus dem Speicher entfernt wird.

Ein weiteres Makro lässt den Balken „wandern“, aktualisiert also seine Darstellung:

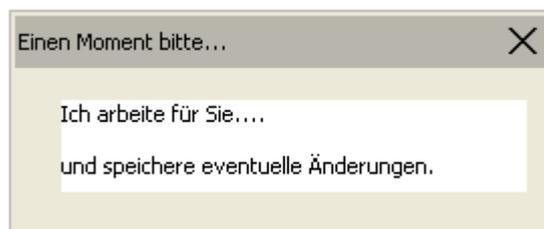
`Status_akt()` im Modul `Helper`. Diese Funktion erwartet zwei Parameter: den im Beschriftungsfeld anzuzeigenden Text sowie die zusätzliche Länge des Statusbalkens (in der Regel eine Stufe):

```

sub Status_akt(stxt as string, iStep as long)
    dim iStepNeu as long
    iStepNeu = oStdlg.getControl("sb").getValue() + iStep
    oStdlg.getControl("txt").text = sTxt
    oStdlg.getControl("sb").setValue(iStepNeu)
end sub

```

Auch der Dialog **Wait** informiert den Benutzer lediglich, dass etwas passiert, wobei dieser komplett statisch aufgebaut ist.



Genutzt wird er beispielsweise beim Speichern der Tabelle und des Dokumentes nach dem Beenden des Belegdialoges.

5.5 Die Makros

Die Makros selbst sind ausführlich mit Kommentaren ausgestattet, so dass sich ein Blick in die Codes durchaus lohnt. Häufig sind auch Modulverweise dort zu finden, wenn eine andere Funktion aufgerufen wird. Dadurch erspart man sich eine lange Suche.

Eine detaillierte Erläuterung der einzelnen Makros wird es in einer späteren Version der Dokumentation an dieser Stelle hier geben.

* * * * *