Program:	***!Konto***
Purpose:	**bank account management**
Version:	**1.0.2a/March 2012**
Tested on:	**BeagleboardxM/RISC OS 5.18 & RPC/RISC OS 4.39**
Author:	**M. R. Beerling**
Date:	**12/3/12**

Summary

A personal finance program:

- Check & control bank statements.
- Generate reports.
- Experiment with data (what if...).
- Transaction data base with search facilities.
- Data plotting using GNU Plot (version 3.71).
- Forward look a head for an account.
- Support for several currencies.
- Open file format for easy conversion to & from other programs.
- Conversion from *!BkMgr* program possible.

!Konto require the !SharedLibs resource from:
http://www.riscos.info/packages/LibraryDetails.html#sharedlibs.

To use the plotting functions *!Konto* requires GnuPlot 3.71 which can be downloaded from:
http://home.tiscali.nl/bobbrand/riscos/gnuplot.html

The Author can be contacted by e-mail on **beerling@online.de,** please use 'Konto' in subject line to get through the spam filters.

Licence

The software is still in development, it is a prototype version. It is not commercial and is supplied "as is", without express or implied warranty. No representations are made about the suitability of this software for any purpose. The author cannot be held responsible or liable for its use or misuse. The software is issued as freeware, it may be freely copied and used but the original author must be acknowledged. Copies of the program must contain all the original files, unchanged and with the original date stamp.

Contents

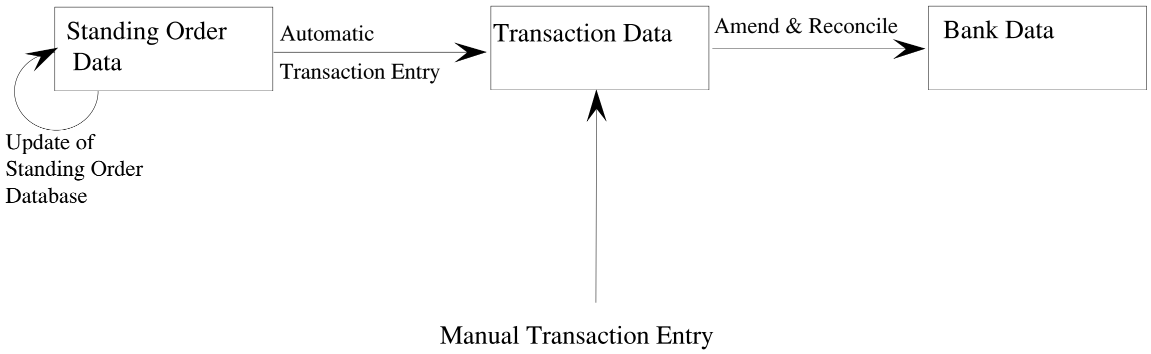
1.Introduction

!Konto is a bank account management program for RISC OS. It is designed to record and predict bank account activity allowing easy analysis of income & expenditure.

!Konto is not an 'online' program. It has nothing to do with the internet or online banking. Transactions, standing orders, accounts and analysis codes are entered into databases. Reports are generated using the data in the databases. *!Konto* was designed to be simple and open source with an open data format allowing future enhancements or easy transfer to another program.

The program was inspired by the Archimedes Bank Manager program from Contex Computing, *!BkMgr* which dates back to the days of the BBC micro (ca 1987). The data from *!BkMgr* can be converted to *!Konto* format using an additional program *!BMConvert* the purpose of which is to convert old transaction, standing order, bank and analysis data to the new format to be used with *!Konto* and to allow migration and easy addition of old transaction data.

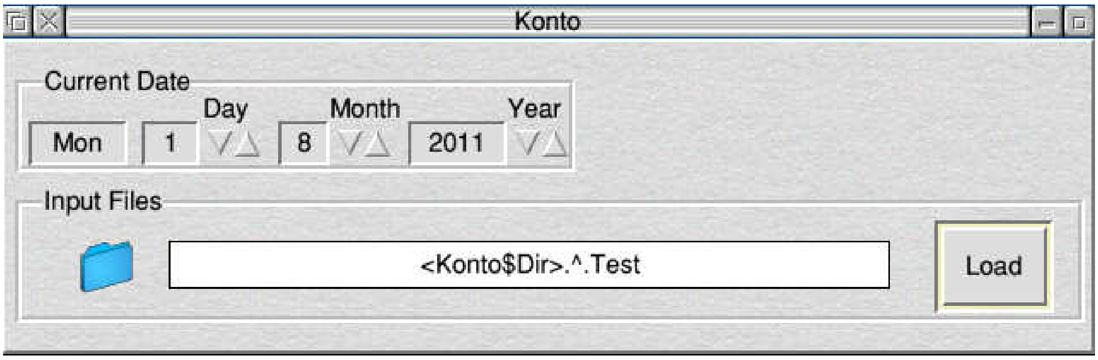
Before using *!Konto* the databases must be set up with initial data or *!BkMgr* converted data can be used. Some thought should be made about what reports are required before adding the analysis codes. It is possible to change codes later but, if a large number of transactions have accrued, this may be difficult. After the initial data has been entered the program automatically adds transactions to the transactions data base using standing orders up to the current date. When a bank statement arrives the transactions can be amended and reconciled so that the bank balance given by the program matches that on the bank statement. In this way the program is used to check the bank statement against expected income & expenditure. However the standing order data base and future transactions can be used by the program to extrapolate the bank balance to a forward date, this is a Bank report. Reports do not change the data in the databases but it is easy to add, remove or change data in the databases and then use the reports to see the affects of any change.



2.Starting *!Konto*

Copy the *!Konto* to a directory on your hard disc then start the program by double clicking with the select mouse button on the filer Icon.

A start up window is displayed with the current date (which can be changed) and the directory where the data files can be found. The source of the input data can be changed by dragging a new directory on to the window or typing the directory name in the writable gadget. The input data files are always in the same directory.



Current Date: Mon, Day 1, Month 8, Year 2011

Input Files: <Konto$Dir>.^.Test

Clicking select on the load button loads the data (if available) or creates new data files if the expected data files cannot be found. The following data files are expected in the given directory: ANALYS, BANK, STO & TRANS.

Two addition files may also be present:

BCKUP an obey file which is called from the program after new data has been written. The contents of the file must be provided by the user.

CHX is the last set of choices used in the program. This is written by *!Konto* when the other data has been saved.

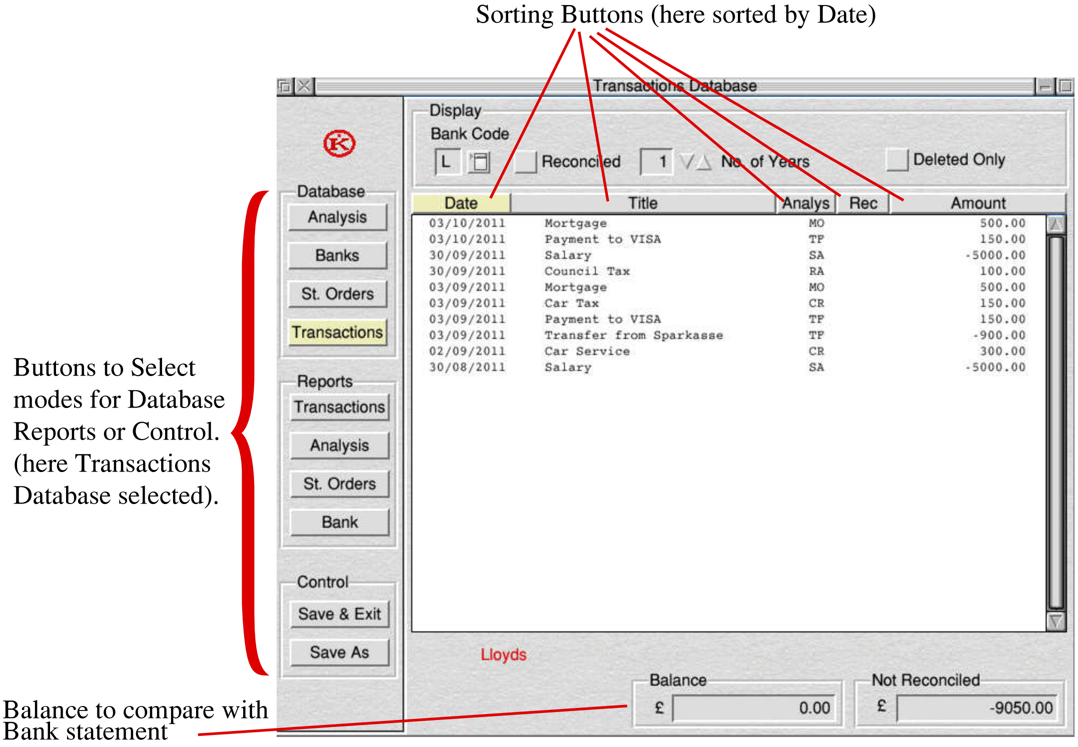
Contents and format of these files is described later.

The date is the current date but another date can be used by clicking on the arrows to changed the day, month or year. Day and month roll over for an easy change from one end to another. The roll over date feature is always available when setting dates in *!Konto*.

The main program window is displayed after the **Load** button has been clicked and data (if any) loaded.

The left hand side on the window contains buttons for the modes described below.

Sorting Buttons (here sorted by Date)



Bank Code: L

☐ Reconciled

1 No. of Years

☐ Deleted Only

Date	Title	Analys	Rec	Amount
03/10/2011	Mortgage	MO		500.00
03/10/2011	Payment to VISA	TF		150.00
30/09/2011	Salary	SA		-5000.00
30/09/2011	Council Tax	RA		100.00
03/09/2011	Mortgage	MO		500.00
03/09/2011	Car Tax	CR		150.00
03/09/2011	Payment to VISA	TF		150.00
03/09/2011	Transfer from Sparkasse	TF		-900.00
02/09/2011	Car Service	CR		300.00
30/08/2011	Salary	SA		-5000.00

Balance: £ 0.00

Not Reconciled: £ -9050.00

Buttons to Select modes for Database Reports or Control. (here Transactions Database selected).

Balance to compare with Bank statement

3.Menu

The mouse menu button is active only when clicked over the main window. Options that can be selected from the menu are:

New Record

Allows the entry of a new data record. Only active in database modes.

CSV Mode

Displays data in the main text window separated with commas. Saved data from the window will be saved in Comma Separated Values (CSV) format otherwise Text files will be saved. CSV format is better for saving to spread sheets or as tables in wordprocessor.

Save

Saves the data displayed in the text window as a file. Either a Text or a CSV file is saved depending on the CSV option selected.

4.Iconbar Menu

Info

Displays information about the program.

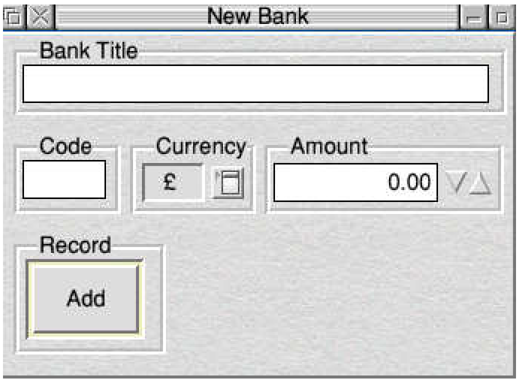
Quit (No Save)

Quits the program without saving any data. No warning is given. Useful when experimenting with the data and the current state of the database is not to be saved.

5.Data base - adding & changing data

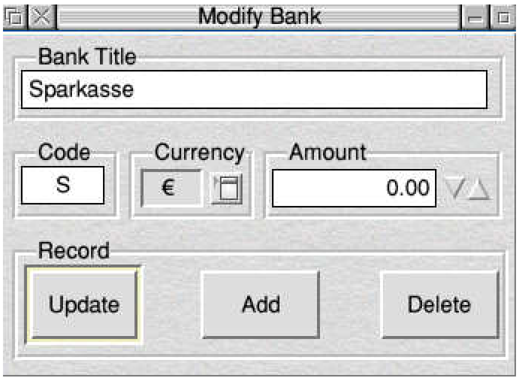
To use *!Konto* data must be added to the at least 2 of the 4 data bases. Firstly banks must be entered (called banks here but could be any account). Next analysis codes are required for sorting & classifying transactions in the the bank accounts. Without any banks and analysis codes the other program mode buttons will be greyed out. Standing orders are entered to act on the bank accounts. Standing orders are also used for direct debits and money coming in to the account. Finally transactions are records of money going into and out of an account. When *!Konto* has been set up properly it should be quick & easy to keep the data up to date.

Generally new data is added by clicking the menu mouse button over the text window and selecting new record from the menu items. A new window will pop up allowing the fields for the item to be filled in and an '**Add**' button to add the new record to the data base.



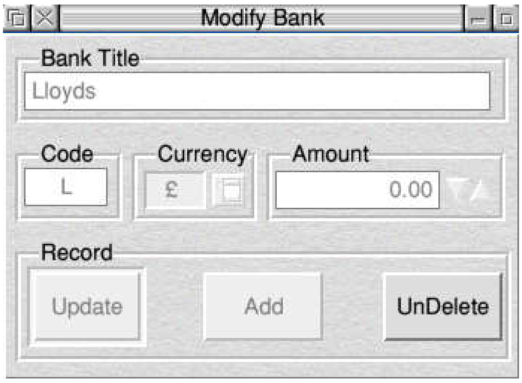
Currency: £
Amount: 0.00

Alternatively an existing record can be clicked with the select button and this data will appear in a new window. With '**Update**' button the data can be changed and the original record overwritten. With the '**Add**' button a new record will be added with this data, The '**Delete**' button deletes the record from the database.



Bank Title: Sparkasse
Code: S
Currency: €
Amount: 0.00

Deleted data can be un-deleted. Clicking on the '**Deleted Only**' option in the display field will show all deleted records. A deleted record can be selected by clicking on it and, in the new window, clicking on the '**unDelete**' button. The 'unDeleted' record will disappear from the Deleted display but will be in the normal display when the '**Deleted Only**' option is clicked off. Deleted records do not play a part in any actions and are not saved when the '**Save & Exit**' or '**Save As'** buttons are clicked on the side panel.



Bank Title: Lloyds

Code: L

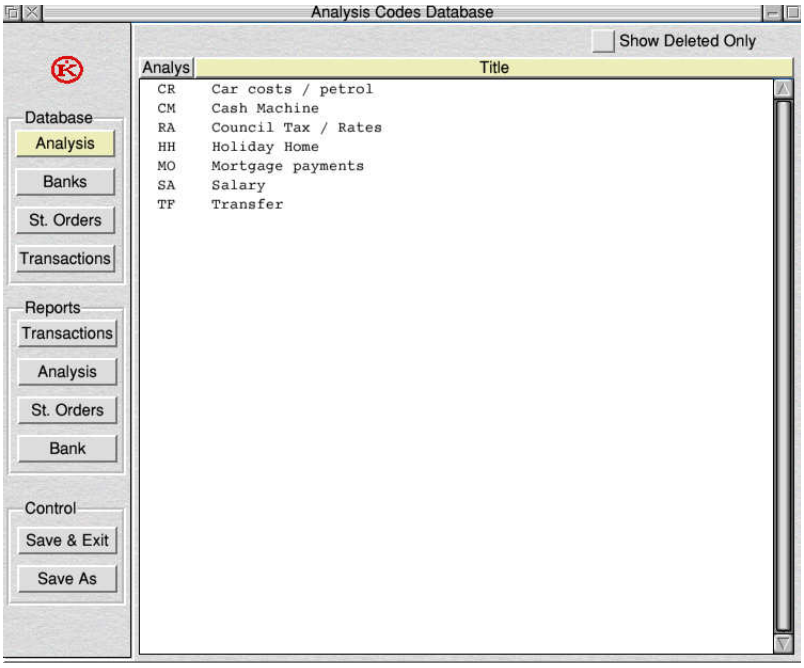
Currency: £

Amount: 0.00

Displayed data can be sorted using the sort buttons above the large text window. Clicking once on the sort button sorts in one direction, clicking again sorts in the reverse direction. The selected sort category is highlighted. A 'Display' area has option buttons to allow a limited amount of filtering as to which records are displayed. Option button '**Deleted Only**' is available in each of the database displays.

A new record can be added to any of the databases by clicking menu over the corresponding window and selecting the option **New Record**. Also available from menu is the **Save** option which saves the displayed data as a text or CSV (Comma Separated Value) file. The **CSV** option, also available from menu, may be useful for dropping the resulting file into a spreadsheet.

Analysis Codes



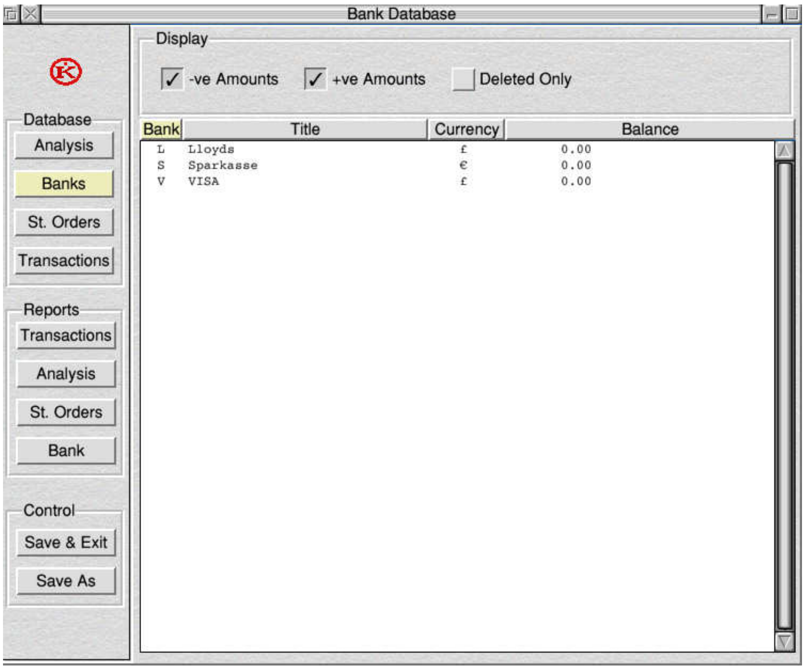
Analys	Title
CR	Car costs / petrol
CM	Cash Machine
RA	Council Tax / Rates
HH	Holiday Home
MO	Mortgage payments
SA	Salary
TF	Transfer

Analysis codes are two letter codes used to tag transaction records for later sorting, filtering & analysis. Each code must be unique. Up to 31 characters text can be used as the title for the code.

In the example data analysis code CR is used for car costs including petrol. All transactions & standing orders for the car are tagged with this code. All the costs for the car can be found in the transactions database & a cost report made.

Analysis codes can be sorted according to code or the code text. The '**Show Deleted Only**' displays just the deleted records (if any).

Banks



☑ -ve Amounts ☑ +ve Amounts ☐ Deleted Only

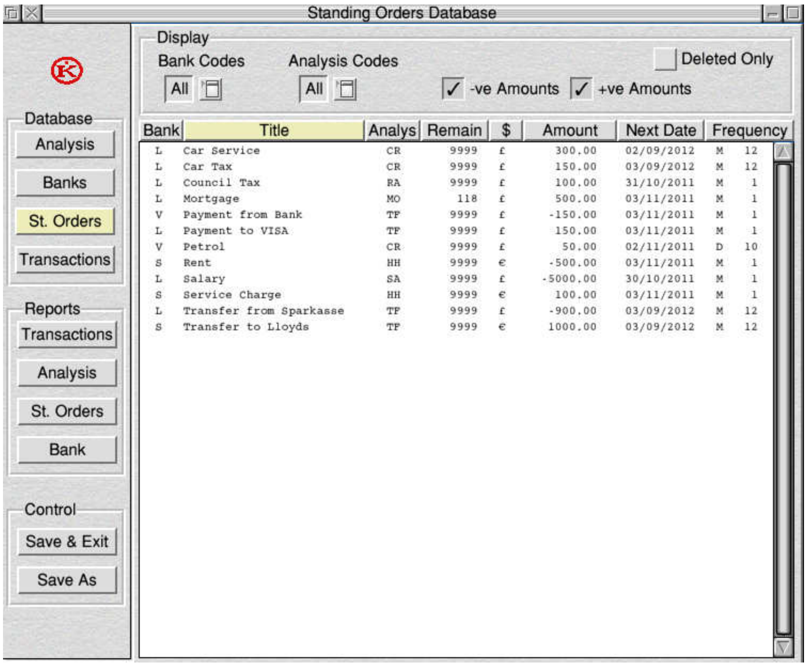
Bank	Title	Currency	Balance
L	Lloyds	£	0.00
S	Sparkasse	€	0.00
V	VISA	£	0.00

Bank codes are single letter codes to identify an account. Each code must be unique. Up to 31 characters text can be used as the title for the bank. Currency of the account and the current balance are all required to complete the record.

Under RISC OS 4.39 the euro sign, €, is not displayed (*!PDF* also can not display the Euro symbol, but *!GView* can). If this is a problem then change the symbol used in file `<Konto$Dir>.Resources.UK.Messages` to EUR. All the currency symbols can be changed, three characters are available for a cuccency so GBP or USD can be used in place of £ or $.

Bank- records can be sorted according to code, title, currency or balance. Filtering can be done on positive and negative balance or deleted only records.

Standing Orders



Bank Codes: All
Analysis Codes: All
☐ Deleted Only
☑ -ve Amounts ☑ +ve Amounts

Bank	Title	Analys	Remain	$	Amount	Next Date	Frequency
L	Car Service	CR	9999	£	300.00	02/09/2012	M 12
L	Car Tax	CR	9999	£	150.00	03/09/2012	M 12
L	Council Tax	RA	9999	£	100.00	31/10/2011	M 1
L	Mortgage	MO	118	£	500.00	03/11/2011	M 1
V	Payment from Bank	TF	9999	£	-150.00	03/11/2011	M 1
L	Payment to VISA	TF	9999	£	150.00	03/11/2011	M 1
V	Petrol	CR	9999	£	50.00	02/11/2011	D 10
S	Rent	HH	9999	€	-500.00	03/11/2011	M 1
L	Salary	SA	9999	£	-5000.00	30/10/2011	M 1
S	Service Charge	HH	9999	€	100.00	03/11/2011	M 1
L	Transfer from Sparkasse	TF	9999	£	-900.00	03/09/2012	M 12
S	Transfer to Lloyds	TF	9999	€	1000.00	03/09/2012	M 12

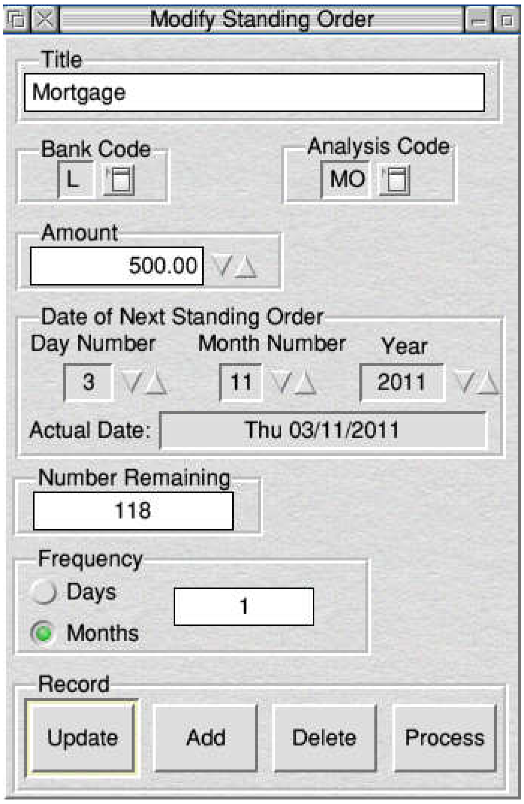
Standing orders are transactions which are performed by a Bank on a regular basis. For *!Konto* the term standing order can be used for any regular transaction into or from a bank account including direct debits. A direct debit for a variable amount can be modelled by a standing order, with a fixed amount, and the actual amount is added when the transaction is reconciled in the Transactions database.

A standing order needs the following information:

Title	title of the transaction e.g. 'Mortgage payment'.
Bank code	the code for the bank involved in the transaction (from the bank database).
Analysis code	the code for classifying the transaction (from the analysis database).
Amount	amount involved. Negative is paying into the account.
Date of Next Standing Order	date when the next transaction will be added to an account.
Number Remaining	number of transactions remaining. Decrements by 1 after the standing order is processed. A value of 9999 does not decrement.
Frequency	number of days or months between each transaction.

The standing order display can be filtered according to Bank codes, Analysis codes, positive (payments out of an account) and negative (payments into an account) amounts. An option to display Deleted only records only is also available. Displayed standing order records can be sorted by Bank,code, Title, Analysis code, number of standing orders remaining, currency, amount, date of next standing order and frequency.

A new standing order can be added by clicking the menu mouse button over the window or an existing record can be changed by clicking on it. A window will appear and the required data can be entered as in the other databases.


Modify Standing Order

Title: Mortgage

Bank Code: L

Analysis Code: MO

Amount: 500.00

Date of Next Standing Order

Day Number: 3

Month Number: 11

Year: 2011

Actual Date: Thu 03/11/2011

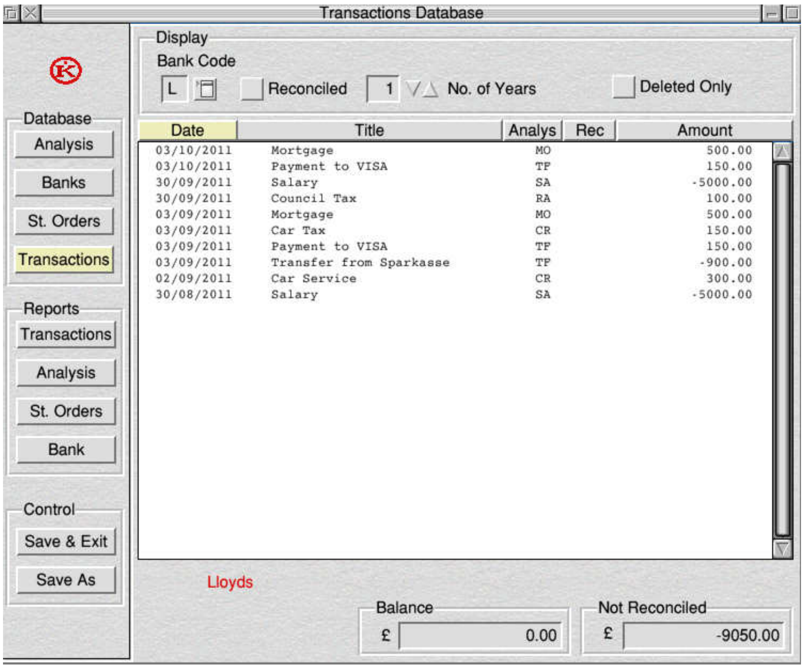
Number Remaining: 118

Frequency:
○ Days
● Months
1

When modifying an existing standing order record there is an extra **Process** button. Normally standing orders are processed automatically on the 'Next Standing Order' date. On this date a transaction is added to the bank account and the 'Next Standing Order' date is increased by the number of days or months given as 'Frequency'. When the **Process** button is clicked the next transaction from the standing order is added to the bank account with the Next Standing Order date although the date for the standing order has not arrived. **Process** is useful for direct debits when the debit amount is known in advanced. The standing order can be processed and the transaction modified for the actual amount rather than waiting until the date when the transaction is added. For example the telephone bill arrives but the amount is not debited from the account until a few weeks later. Rather than wait for the transaction from the standing order to be added to the account the transaction can be added with the **Process** button. The exact amount for the transaction can be modified in the transaction database.

The standing order model in *!BkMgr* has a bug. If you have a monthly standing order for the 31st of the month then, for months which only have 30 days, the Next Standing Order date will get set to the 30th. From then onwards the transaction will be added on the 30th (even for months with 31 days). Worst still is February which has, mostly, 28 days. When February is reached the Next Standing Order date moves down to the 28th and remains there for subsequent months. To overcome the *!BkMgr* bug an artificial date is given as the 'Date of Next Standing Order' . The date given may not be a date that can exist. For example the 30th February can be given which is a date that will never exist. The actual date used for the next date is displayed as 28th (or 29th) February. If the given date is after the last date of the month then *!Konto* moves the date back to the last day of the month. In the above example, if the standing order is monthly, the dates will progress on to the 30th March, 30th April and so on. This is the only occasion in *!Konto* that an artificial date is used.

Transactions



Bank Code: L

☐ Reconciled

1 No. of Years

☐ Deleted Only

Date	Title	Analys	Rec	Amount
03/10/2011	Mortgage	MO		500.00
03/10/2011	Payment to VISA	TF		150.00
30/09/2011	Salary	SA		-5000.00
30/09/2011	Council Tax	RA		100.00
03/09/2011	Mortgage	MO		500.00
03/09/2011	Car Tax	CR		150.00
03/09/2011	Payment to VISA	TF		150.00
03/09/2011	Transfer from Sparkasse	TF		-900.00
02/09/2011	Car Service	CR		300.00
30/08/2011	Salary	SA		-5000.00

Lloyds

Balance £ 0.00

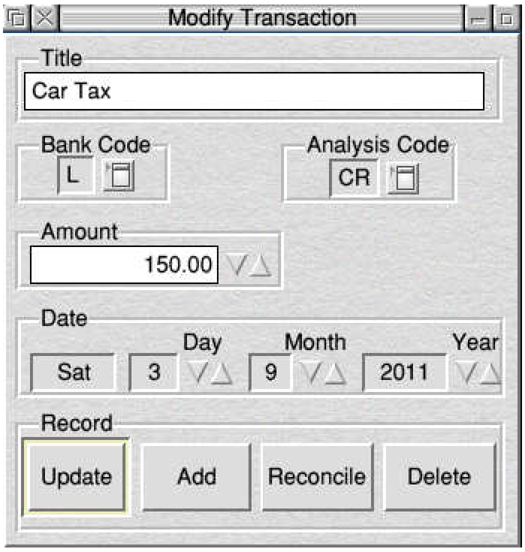
Not Reconciled £ -9050.00

Transactions are records of money paid into or out of a bank account. Positive transaction values are paid out of the account and negative values payed into the account. All transactions are stored in a single database but to be useful they are displayed according to a bank account. The transaction database corresponds to the Amend and Reconcile function in *!BkMgr*. Transactions are checked against the current bank statement, amended and reconciled as appropriate.

A reconciled transaction has been verified against a bank statement and the amount added to the bank account. Reconciled transactions are usually filtered out of the display since they have been verified as correct. Only unreconciled transactions are normally displayed since they have to be checked against a bank statement however with the **Reconciled** option reconciled records will also be displayed.

The **No. of Years** number selection allows transaction records to be displayed for a given number of years. A text window which displays a large number of records is slow to update, it is better to limit the number of transaction records to be displayed.

Only unreconciled transactions can be deleted and, as with the other databases, a **Deleted Only** selection is available to display only deleted transactions.



Title: Car Tax
Bank Code: L
Analysis Code: CR
Amount: 150.00
Date: Sat
Day: 3
Month: 9
Year: 2011

Transactions can be modified by clicking select on the transaction record or a new one added by clicking menu over the text window and selecting **New Record**. A transaction must have a Title, Bank code, Analysis code, amount and a date.

For a new transaction there is only the **Add** button to add the transaction data as a record to the transaction database. For an existing transaction there are four buttons in the 'Modify Transaction' window. **Update** overwrites the existing transaction record with the data in the 'Modify Transaction' window. **Add** keeps the existing transaction record and adds a new one with the data in the 'Modify Transaction' window. **Delete** removes the transaction record from the transaction database, the data in the 'Modify Transaction' window is ignored. **Reconcile** has a special meaning. When the transaction is checked against a bank statement (and perhaps corrected) the **Reconcile** button can be clicked. The amount of the transaction is subtracted from the the bank account and the transaction is flagged as reconciled so that it doesn't, normally, appear in the transaction database display. The bank balance displayed by *!Konto* should agree with that in the bank statement.

Reconciled transactions can be displayed in the transaction database window by selecting the **Reconciled** display option. They can be modified by clicking on them, like unreconciled transactions, but the only option is to unreconcile (**unRec.**) the transaction. When a reconciled transaction is unreconciled then the amount is added to the bank account and the record becomes a normal unreconciled transaction (i.e. the reconcile process is reversed). The process to unreconcile a transaction is to correct errors and is not part of the normal flow of actions.

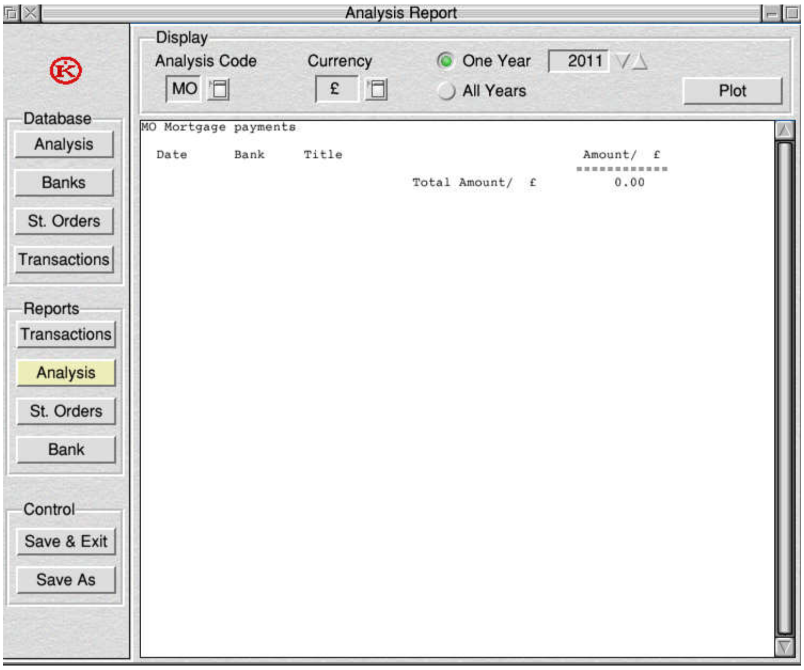
6.Reports - extracting information

The database part of *!Konto* is for adding and changing data in the databases. The reports section is about viewing the data in different ways either for an overview or for inclusion in other documents such as accounts or tax returns. Reports do not change the data in the databases.

Most report windows have a **Plot** button. Unless the filer has seen *!GnuPlt371* this button will be greyed out. GnuPlot V3.71 has been tested with *!Konto* and found to work. It may be that later versions will also work but earlier versions did not work.

Displayed data can be saved as a text or CSV (Comma Separated Value) file by clicking menu over the window and selecting the **Save** option**.** The file icon must be dragged to a file viewer. The **CSV** option is also availed from menu and is useful for dropping the resulting file into a spreadsheet. Option New Record is not available for reports and is greyed out.

Analysis



Analysis Code: MO
Currency: £
● One Year 2011
○ All Years

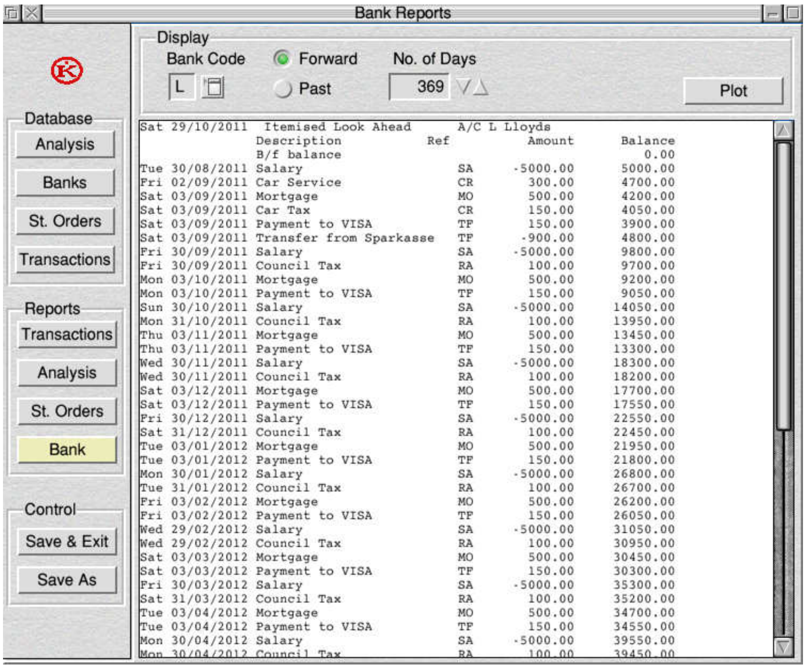
```
MO Mortgage payments

  Date       Bank      Title                              Amount/  £
                                                        ============
                                   Total Amount/  £          0.00
```

An Analysis report lists and adds up all transactions for a given **Analysis code** and **Currency** and **Year**. If the option **All Years** is clicked then the list and sum is for all years where transaction exist. The **Plot** button takes the displayed data and produces a simple plot of amount against date. The purpose is an overview of income and outgoings for an analysis code over a year or more years.

The displayed data can be saved as a text or CSV file by clicking menu over the window.

Banks



Bank Code: L

● Forward
○ Past

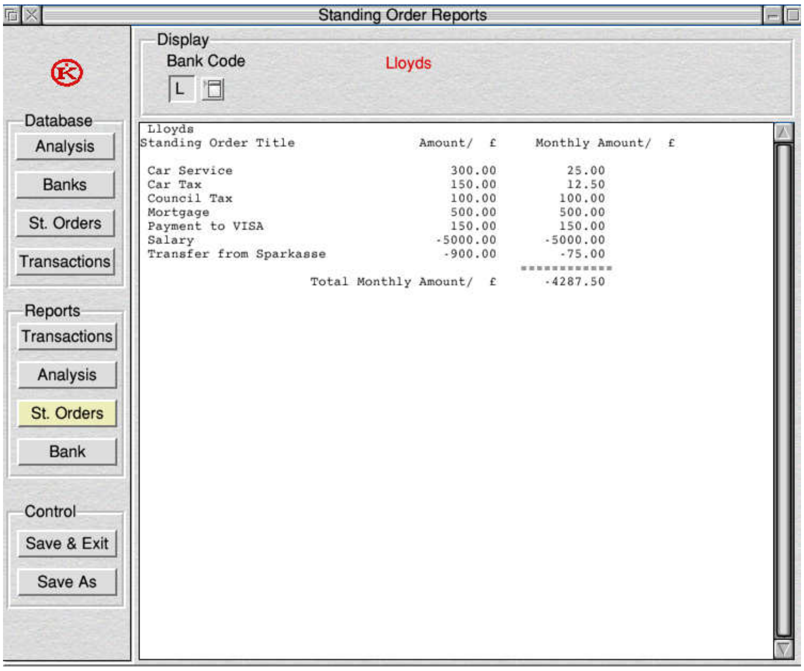
No. of Days: 369

```
Sat 29/10/2011  Itemised Look Ahead      A/C L Lloyds
               Description           Ref           Amount     Balance
               B/f balance                                       0.00
Tue 30/08/2011 Salary                  SA        -5000.00     5000.00
Fri 02/09/2011 Car Service             CR          300.00     4700.00
Sat 03/09/2011 Mortgage                MO          500.00     4200.00
Sat 03/09/2011 Car Tax                 CR          150.00     4050.00
Sat 03/09/2011 Payment to VISA         TF          150.00     3900.00
Sat 03/09/2011 Transfer from Sparkasse TF         -900.00     4800.00
Fri 30/09/2011 Salary                  SA        -5000.00     9800.00
Fri 30/09/2011 Council Tax             RA          100.00     9700.00
Mon 03/10/2011 Mortgage                MO          500.00     9200.00
Mon 03/10/2011 Payment to VISA         TF          150.00     9050.00
Sun 30/10/2011 Salary                  SA        -5000.00    14050.00
Mon 31/10/2011 Council Tax             RA          100.00    13950.00
Thu 03/11/2011 Mortgage                MO          500.00    13450.00
Thu 03/11/2011 Payment to VISA         TF          150.00    13300.00
Wed 30/11/2011 Salary                  SA        -5000.00    18300.00
Wed 30/11/2011 Council Tax             RA          100.00    18200.00
Sat 03/12/2011 Mortgage                MO          500.00    17700.00
Sat 03/12/2011 Payment to VISA         TF          150.00    17550.00
Fri 30/12/2011 Salary                  SA        -5000.00    22550.00
Sat 31/12/2011 Council Tax             RA          100.00    22450.00
Tue 03/01/2012 Mortgage                MO          500.00    21950.00
Tue 03/01/2012 Payment to VISA         TF          150.00    21800.00
Mon 30/01/2012 Salary                  SA        -5000.00    26800.00
Tue 31/01/2012 Council Tax             RA          100.00    26700.00
Fri 03/02/2012 Mortgage                MO          500.00    26200.00
Fri 03/02/2012 Payment to VISA         TF          150.00    26050.00
Wed 29/02/2012 Salary                  SA        -5000.00    31050.00
Wed 29/02/2012 Council Tax             RA          100.00    30950.00
Sat 03/03/2012 Mortgage                MO          500.00    30450.00
Sat 03/03/2012 Payment to VISA         TF          150.00    30300.00
Fri 30/03/2012 Salary                  SA        -5000.00    35300.00
Sat 31/03/2012 Council Tax             RA          100.00    35200.00
Tue 03/04/2012 Mortgage                MO          500.00    34700.00
Tue 03/04/2012 Payment to VISA         TF          150.00    34550.00
Mon 30/04/2012 Salary                  SA        -5000.00    39550.00
Mon 30/04/2012 Council Tax             RA          100.00    39450.00
```

The Bank Report corresponds to the forward lookahead from *!BkMgr*. It takes the existing unreconciled transactions for a bank (**Bank Code**) and, using the standing order database, extrapolates the account forward in time (**No of Days**). The purpose being to see when or if the account goes negative and so to take corrective action in advance. Its also good for financial modelling, that is changing the timing or amount of standing orders and examining the effect on a bank account. Like a weather forecast this modelling becomes less accurate the further ahead the forecast goes.

The option **Forward** extrapolates the bank account data forward in time but **Past** uses actual transaction data to look back in to the past. The balance against date can be plotted using the **Plot** button.

Standing Orders



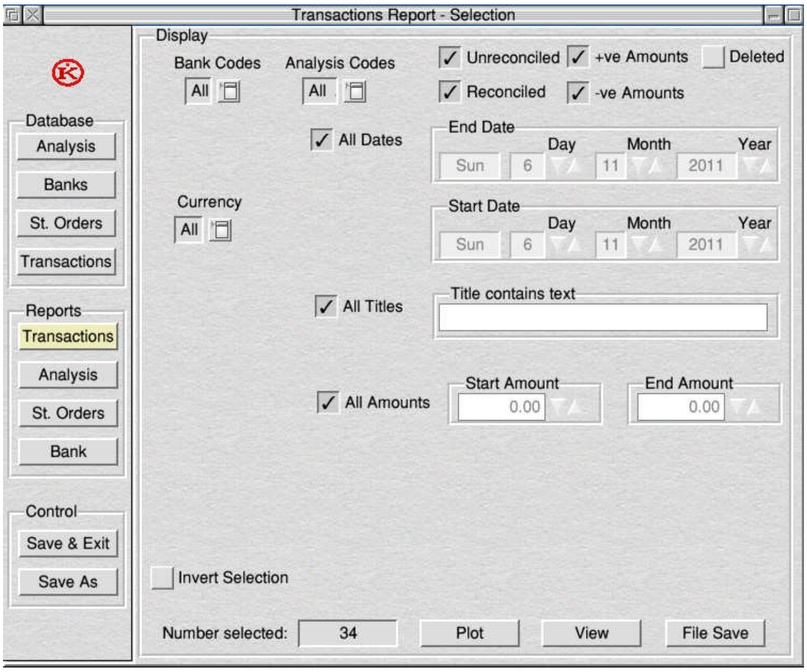
Bank Code: L — Lloyds

```
 Lloyds
Standing Order Title                      Amount/  £      Monthly Amount/  £

 Car Service                               300.00         25.00
 Car Tax                                   150.00         12.50
 Council Tax                               100.00        100.00
 Mortgage                                  500.00        500.00
 Payment to VISA                           150.00        150.00
 Salary                                  -5000.00      -5000.00
 Transfer from Sparkasse                  -900.00        -75.00
                                                    ============
                      Total Monthly Amount/  £         -4287.50
```

The standing order report lists all active standing orders for a bank account (**Bank Code**). The amount for each standing order is normalized to a monthly sum and then added together. The purpose is to see if the standing orders send the account into negative.

Transactions



Bank Codes: All
Analysis Codes: All
☑ Unreconciled ☑ +ve Amounts ☐ Deleted
☑ Reconciled ☑ -ve Amounts
☑ All Dates
End Date: Day Sun 6, Month 11, Year 2011
Start Date: Day Sun 6, Month 11, Year 2011
Currency: All
☑ All Titles
Title contains text:
☑ All Amounts
Start Amount: 0.00
End Amount: 0.00
☐ Invert Selection
Number selected: 34

The transactions report is used to search the transaction database for transactions which match a given criteria. It has two parts, a selection part (shown above) and a view part (shown when the **View** button is pressed).

Transactions found are those that match all the criteria.

Bank Codes	Either all bank codes (default) or transactions with the bank code are found.
Analysis Codes	Either all analysis codes (default) or transactions with the analysis code are found.
Unreconciled	Unreconciled transactions (default) are found.
Reconciled	Reconciled transactions (default) are found.
+ve Amounts	Transactions with positive amounts (default) are found.
-ve Amounts	Transactions with negative amounts (default) are found.
Deleted	Deleted transactions are found (not default)
All Dates	Transactions with all dates are found (default) otherwise the **Start & End dates** are used.
End Date	Greyed out unless **All Dates** is not selected. The last date of transactions found.
Start Date	Greyed out unless **All Dates** is not selected. The first date of transactions found.
Currencies	Either all currencies (default) or transactions with the currency code are found.
All Titles	Transactions with all titles are found (default) otherwise the **Title contains Text** is used.
Titles contain Text	Greyed out unless **All Titles** is not selected. Transactions containing the given text within the title are found.
All Amounts	Transactions with all amounts are found (default) otherwise the **Start & End Amount** are used.
Start Amount	Greyed out unless **All Amounts** is not selected. The first amount for transactions found.
End Amount	Greyed out unless **All Amounts** is not selected. The last amount for transactions found.
Invert Selection	Inverts the selection from the other selection criteria. If, say, 100 transactions out of 500 have been selected then, with **Invert Selection** clicked, the other 400 transactions will be selected.

Note that the bank and analysis codes are selected not from the analysis or bank database but from all analysis and bank codes present in all the transactions. This way orphaned transactions can be found.

The displayed **Number selected** is the number of transactions selected matching the criteria given.

Plot button plots the transaction amounts against date (although this may not always be meaningful).
View button changes to 'Transactions Report - View' window which displays the selected transactions. In this window the transactions can then be sorted according to **Date**, **Bank Code**, **Title**, **Analysis Code**, **Reconciled**, **Currency** and **Amount**. To return to the 'Transactions Report - Selection' window the Reports side button **Transactions** must be clicked.

File Save button saves the selected transactions in the same format as the TRANS file read in by *!Konto*. The purpose is to aid archiving of old transaction data and so keeping the TRANS file small, although this should not be necessary.

7.Program Control

After the database options and the reporting options program control is the next section on the left side button bar. Two buttons are available.

Save & Exit

Clicking on the **Save & Exit** button will save the data back to the files from where the data was read and exit the program. In normal use when transactions have been modified or new ones added this is the quickest way to finish with the program. The directory where the data is saved is the same as used to read the data unless the path has been changed using the **Save As** button (see below).

Save As

The **Save As** button allows the user to change the directory path where the current data will be saved. After clicking on the button a window appears with the current path in a writable text gadget. The current path can be changed by clicking on the text and writing another path, or by dragging & dropping a directory icon from a file viewer on to the window. If the given directory doesn't exist then an error message is issued. The purpose of this option is to save temporary data. After being changed *!Konto* will use the new path for the the **Save & Exit** option.

8.Contents & Format of Data files

An open source program should also have an open file format. This helps the user move data to other programs or edit the data with a standard editor.

For this reason all the data files used by *!Konto* are standard text files which can be examined and edited with an editor.

All the database files contain a header to establish the identity of the file, then each line is one data record.

A choices file is also written to the data directory which contain the choices for the data. A settings file is written to the same directory as *!Konto* and contains the last settings when the program was used.

File ANALYS

This file contains the analysis database.

The first line is a header to identify the file. Each line after the header contains one record of data. For each analysis record the first two characters are the Analysis Code and the following 31 characters are the analysis title. The record order within the file is not important.

```
#!Konto_ANALYS_V1
SASalary
CRCar costs / petrol
MOMortgage payments
CMCash Machine
RACouncil Tax / Rates
HHHoliday Home
TFTransfer
```

File BANK

This file contains the bank database.

The first line is a header to identify the file. Each line after the header contains one record of data. For each bank record the first character is the Bank Code, the following 31 characters are the bank title followed by 3 characters for the bank currency. After the currency is an integer which is the current bank balance in pence or cents (depending on the currency). The record order within the file is not important.

```
#!Konto_BANK_V2
LLloyds                          £  0
VVISA                            £  0
SSparkasse                       €  0
```

File TRANS

This file contains the transaction database

The first line is a header to identify the file. Each line after the header contains one record of data. For each transaction record the first character is the Bank Code, the next two characters are the Analysis Code, the following 31 characters are the transaction title followed by 10 characters for the date. After the date is an integer which is the transaction amount in pence or cents (depending on the currency). The order of the transactions is not important (although *!Konto* saves them in date order).

```
#!Konto_TRANS_V1
TTTPayment in                  01/08/2011 -100000
TTTPayment Out                 01/08/2011 10000
VTFPayment from Bank           03/08/2011 -15000
```

File STO

This file contains the standing order database

The first line is a header to identify the file. Each line after the header contains one record of data. For each standing order record the first character is the Bank Code, the next two characters are the Analysis Code, the following 31 characters are the standing order title followed by 10 characters for the date. After the date are 3 integers which are the number of standing orders left to perform (9999 means infinity), the amount in pence or cents (depending on the currency) and the frequency. A frequency over 1000 means monthly, under 1000 means daily. For example 1001 means every month and 1012 means every 12 months. 10 means every ten days. The record order within the file is not important.

```
#!Konto_STO_V1
LSASalary                         30/08/2011       9999   -500000       1001
LCRCar Service                    02/09/2011       9999     30000       1012
VCRPetrol                         03/09/2011       9999      5000         10
LMOMortgage                       03/09/2011        120     50000       1001
LRACouncil Tax                    31/09/2011       9999     10000       1001
SHHRent                           03/09/2011       9999    -50000       1001
SHHService Charge                 03/09/2011       9999     10000       1001
```

File CHX

this is the choices file. The first integer is the days to look ahead used for the bank report. The next line contains two integers for the position of the window. The last line is the preferred bank code to be used at startup.

```
369
856 1488
L
```

File BCKUP

A user supplied obey file is called just before the data are saved. This file is called BCKUP and it must be in the same directory as TRANS, BANK, ANALYS, STO and CHX. It should contain commands to back up the existing data files before they are over written by new data. If no BCKUP file is present then it is ignored.

An example might be:

```
copy <Konto$Dir>.^.Example.TRANS  <Konto$Dir>.^.Example.Backup.* FQ~C
copy <Konto$Dir>.^.Example.BANK   <Konto$Dir>.^.Example.Backup.* FQ~C
copy <Konto$Dir>.^.Example.ANALYS <Konto$Dir>.^.Example.Backup.* FQ~C
copy <Konto$Dir>.^.Example.STO    <Konto$Dir>.^.Example.Backup.* FQ~C
copy <Konto$Dir>.^.Example.CHX    <Konto$Dir>.^.Example.Backup.* FQ~C
```

Which copies all files in the DATAFILES directory to a directory called Backup. Of course copying the data to another storage device would be better.

File KontoSettings

This file is saved within the directory of the *!Konto* program and contains the program settings (as opposed to the user choices CHX). Currently it contains just one line and that is the last directory used for the input data. The location and name of this file can be changed by editing the Messages file in `<Konto$Dir>.Resources.UK.Messages`.

```
<Konto$Dir>.^.Example
```

9.Notes on Compiling

A user might like to change the text display of !Konto or add new features. All source code is provided along with make files. *!Konto* was written in C++ and uses the gcc version 4.1.1-release 2 compiler. It is linked with OSLib version 7 to generate an ELF image (hence *!Konto* require the *!SharedLibs* resource from http://www.riscos.info/packages/LibraryDetails.html#sharedlibs.)

A few symbols must be defined before the program can be compiled. For OSLib the symbols OSLib$Dir, OSLib$Path, OSLibInclude$Path, OSLibSupport$Dir & OSLibSupport$Path must be defined. The definition depends on where the library is on the compiling system but the following is an example.

```
Set OSLib$Dir SCSI::xM_32GB.$.Programming.OSLib
Set OSLib$Path SCSI::xM_32GB.$.Programming.OSLib.
Set OSLibInclude$Path <OSLib$Dir>.
Set OSLibSupport$Dir SCSI::xM_32GB.$.Programming.OSLibSupport
Set OSLibSupport$Path <OSLibSupport$Dir>.
```

Symbols for the *!Konto* libraries must be defined as follows:

```
Set KontoLib$Path <Konto$Dir>.Library.KontoLib.
Set KontoWLib$Path <Konto$Dir>.Library.KontoWLib.
```

and for *!BMConvert*:

```
Set BMDLib$Path <BMConvert$Dir>.BMDLib.
```

Make sure symbols <Konto$Dir> & <BMConvert$Dir> have been set correctly (they can be set by running each program).

When the symbols have been defined the *!Konto* or *!BMConvert* application can be opened (Shift-click on the application filer icon) and `Menu->Set directory` called to set the current directory. Then in a terminal window (opened with sufficient memory) the command '`make`' can be given.

AcornC/C++

With a suitable version of OSLib (e.g. version 6.9) it is possible to compile *!Konto* with Acorns C/C++ compiler. The library symbols must be set as for GCC. Additionally a small change is required to file `OSLibSupport:Systypes.h`. Line 80 must be replaced.

Original:

```
#if !defined( _kernel_h )
```

Replace with:

```
#if !defined( _kernel_h ) && !defined( __kernel_h ) && !defined( __KERNEL_H )
```

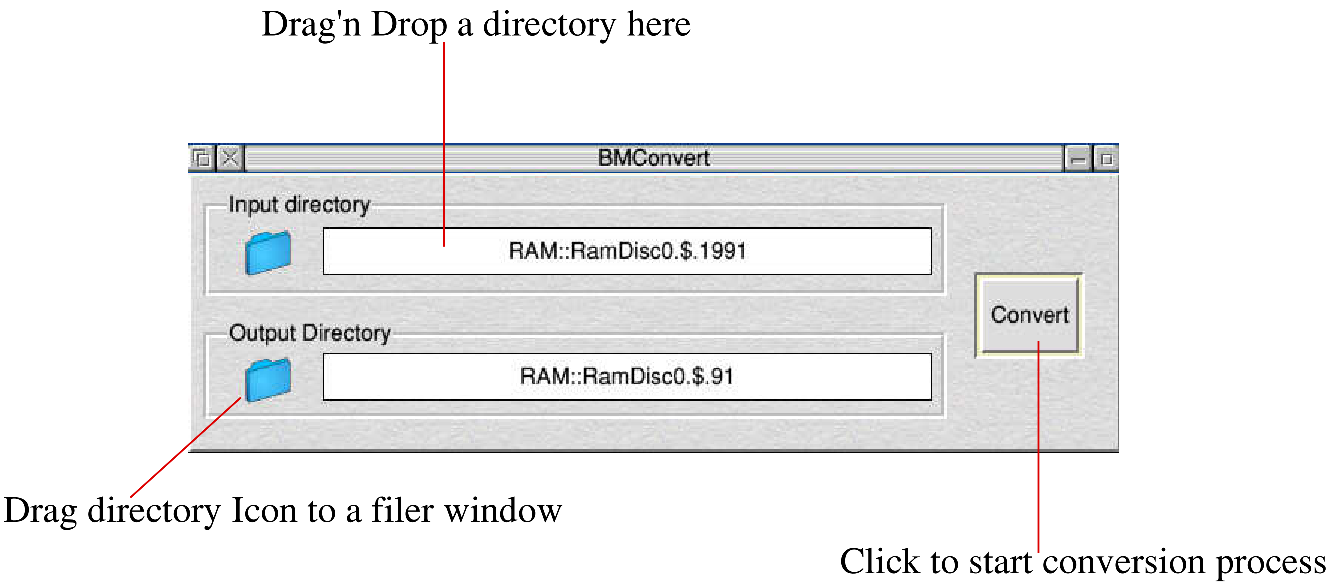
The amu make files are supplied as `Makefile_amu` and must be renamed to `Makefile` for the amu utility to work. Also all '`cc`' directorys must be renamed to '`c++`'. The corresponding make command in the task window is '`amu`'.

10.File Conversion from *!BkMgr* Format

!Konto was designed to be a replacement for *!BkMgr* and one requirement was to be able to convert the older *!BkMgr* files (BMDANALYS, BMDBANK, BMDSYSTEM & BMDTRANS) to *!Konto* data files (ANALYS, BANK, STO & TRANS).

A separate application is provided to convert data files from the *!BkMgr* format to that used by *!Konto*. This is a simple utility application called *!BMConvert*. It is expected to be used once (if at all) and then can be archived away.

When started *!BMConvert* opens a single window. No Iconbar Icon is produced. Closing the window exits the program.


Drag'n Drop a directory here
BMConvert
Convert
Drag directory Icon to a filer window
Click to start conversion process

Input directory: RAM::RamDisc0.$.1991

Output Directory: RAM::RamDisc0.$.91

The user must drag & drop a directory on to the *!BMConvert* window. The directory contains the *!BkMgr* files BMDANALYS, BMDBANK, BMDSYSTEM and BMDTRANS which will be converted. An output directory name must be given where the converted !*Konto* files ANALYS, BANK, STO and TRANS will be written.

The program converts only reconciled transactions. If the converted transactions (in file TRANS) need to be concatenated with other transactions (e.g. because they have been archived) then a simple text editor can perform this task. Note that the file header must appear only once on the first line and that the order of the transactions is not important. If older analysis or bank records are concatenated then care should be taken that only unique analysis or bank codes are present.

11.Files Released

```
*info !Konto.*
!Boot          WR/     Obey      20:26:15 06-Aug-2009   57  bytes
!Help          WR/     Obey      16:48:06 09-Oct-2011  163  bytes
!Run           WR/     Obey      12:36:36 09-Oct-2011 1775  bytes
!RunImage      WR/R    ELF       15:15:29 18-Feb-2012  399 kbytes
!Sprites       WR/     Sprite    17:59:45 28-Sep-2011 1928  bytes
cc             D/      Directory 17:56:47 08-Jan-2011    0  bytes
GNUPlot/cmd    WR/     Text      19:00:25 30-Jun-2010  292  bytes
h              D/      Directory 17:56:47 08-Jan-2011    0  bytes
KontoSettings  WR/     Text      12:54:09 01-Jan-2012   22  bytes
Library        D/      Directory 17:56:49 08-Jan-2011    0  bytes
Makefile       WR/     Text      14:56:34 02-Jul-2011 1373  bytes
Makefile_amu   WR/     Text      12:40:01 01-Jan-2012    8 kbytes
o              D/      Directory 12:44:57 01-Jan-2012    0  bytes
Resources      D/      Directory 09:25:43 08-Oct-2011    0  bytes

*info !Konto.cc.*
!Konto       WR/     Text      12:00:05 01-Jan-2012   17 kbytes

*info !Konto.h.*
KONTO_T_     WR/     Text      16:48:19 08-Jul-2011 1775  bytes

*info !Konto.Library.*
KontoLib     D/      Directory 17:56:49 08-Jan-2011    0  bytes
KontoWLib    D/      Directory 17:56:51 08-Jan-2011    0  bytes

*info !Konto.Library.KontoLib.*
cc           D/      Directory 17:56:49 08-Jan-2011    0  bytes
h            D/      Directory 17:56:50 08-Jan-2011    0  bytes
Makefile     WR/     Text      18:12:05 13-Mar-2011 2794  bytes
Makefile_amu WR/     Text      11:49:25 01-Jan-2012   13 kbytes
o            D/      Directory 17:56:51 08-Jan-2011    0  bytes

*info !Konto.Library.KontoLib.cc.*
ANALYS       WR/     Text      11:34:26 01-Jan-2012   10 kbytes
ANALYS_LST   WR/     Text      20:25:27 15-Jul-2009 1161  bytes
BANK         WR/     Text      11:34:33 01-Jan-2012   13 kbytes
BANK_LST     WR/     Text      20:49:26 21-Jul-2009 1145  bytes
CHX          WR/     Text      17:30:04 14-Nov-2010 2469  bytes
DATE         WR/     Text      16:21:06 31-Dec-2011    7 kbytes
ERR          WR/     Text      09:03:24 17-Apr-2011    4 kbytes
FILE         WR/     Text      16:39:38 27-Sep-2004 4068  bytes
FLA          WR/     Text      14:42:39 18-Feb-2012    9 kbytes
LST_T        WR/     Text      16:24:55 27-Sep-2004 1395  bytes
SORT         WR/     Text      14:06:11 14-Sep-2008 1380  bytes
STO          WR/     Text      10:14:24 31-Dec-2011   13 kbytes
STO_LST      WR/     Text      18:01:41 04-Jul-2008 1148  bytes
TRANS        WR/     Text      14:28:07 18-Feb-2012   16 kbytes
TRANS_LST    WR/     Text      16:50:27 08-Aug-2010 1160  bytes

*info !Konto.Library.KontoLib.h.*
ANALYS_      WR/     Text      18:02:41 13-Mar-2011 2280  bytes
ANALYS_LST_  WR/     Text      19:02:05 11-Jul-2008 1111  bytes
ANALYS_T_    WR/     Text      14:13:50 29-Apr-2011 2759  bytes
BANK_        WR/     Text      09:41:09 02-Jul-2011 2688  bytes
BANK_LST_    WR/     Text      16:49:43 06-Jul-2008 1099  bytes
BANK_T_      WR/     Text      09:14:04 17-Apr-2011 2673  bytes
CHX_         WR/     Text      15:29:44 04-Jun-2010 1399  bytes
DATE_        WR/     Text      18:03:35 11-Aug-2011 2625  bytes
ERR_         WR/     Text      16:56:58 02-May-2010 2699  bytes
FILE_        WR/     Text      16:40:29 27-Sep-2004 2209  bytes
FLA_         WR/     Text      10:47:21 20-Jun-2010 2525  bytes
FLA_T_       WR/     Text      08:59:01 17-Jun-2010  927  bytes
LST_B_       WR/     Text      21:00:50 31-Dec-2011   16 kbytes
```

```
LST_S_       WR/     Text      21:12:11 31-Dec-2011    7 kbytes
LST_T_       WR/     Text      17:04:50 31-Dec-2011 1994  bytes
SORT_        WR/     Text      14:03:34 14-Sep-2008 1104  bytes
SORT_T_      WR/     Text      17:00:36 17-Jan-2010 1027  bytes
STD_T_       WR/     Text      19:08:16 01-Oct-2010 1735  bytes
STO_         WR/     Text      17:40:21 11-Aug-2011 2379  bytes
STO_LST_     WR/     Text      20:22:21 26-Jun-2008 1104  bytes
STO_T_       WR/     Text      16:32:19 12-Aug-2011 3564  bytes
TRANS_       WR/     Text      17:21:33 06-Nov-2011 2579  bytes
TRANS_LST_   WR/     Text      14:12:05 18-May-2008 1112  bytes
TRANS_T_     WR/     Text      15:04:29 02-Jul-2011 3192  bytes

*info !Konto.Library.KontoWLib.*
cc           D/      Directory 17:56:51 08-Jan-2011    0  bytes
h            D/      Directory 17:56:52 08-Jan-2011    0  bytes
Makefile     WR/     Text      14:37:53 02-Jan-2012    5 kbytes
Makefile_amu WR/     Text      13:06:40 01-Jan-2012   47 kbytes
o            D/      Directory 17:56:53 08-Jan-2011    0  bytes

*info !Konto.Library.KontoWLib.cc.*
AMREW        WR/     Text      15:03:56 18-Feb-2012   31 kbytes
ANALYSRW     WR/     Text      18:28:07 17-Jan-2012   20 kbytes
ANALYSW      WR/     Text      11:38:43 01-Jan-2012   18 kbytes
BANKW        WR/     Text      11:39:34 01-Jan-2012   20 kbytes
CSVW         WR/     Text      12:22:01 31-Dec-2011   11 kbytes
DATEW        WR/     Text      13:34:52 27-Mar-2011    5 kbytes
FLAW         WR/     Text      12:37:32 01-Jan-2012   14 kbytes
PLOTW        WR/     Text      13:05:05 01-Jan-2012    6 kbytes
STD          WR/     Text      18:41:19 08-Jan-2011 2574  bytes
STORW        WR/     Text      11:43:03 01-Jan-2012   11 kbytes
STOW         WR/     Text      18:20:25 17-Jan-2012   31 kbytes
TRANSW       WR/     Text      12:31:32 01-Jan-2012   20 kbytes
TRANSW_V     WR/     Text      14:49:43 18-Feb-2012   12 kbytes

*info !Konto.Library.KontoWLib.h.*
AMREW_       WR/     Text      16:35:53 04-Jun-2010 1124  bytes
AMREW_T_     WR/     Text      17:34:52 04-Feb-2011 2095  bytes
ANALYSRW_    WR/     Text      17:08:51 30-Apr-2011  885  bytes
ANALYSRW_T_  WR/     Text      08:39:33 07-May-2011 1327  bytes
ANALYSW_     WR/     Text      13:55:41 31-Jul-2011  896  bytes
ANALYSW_T_   WR/     Text      09:47:30 17-Oct-2010 1513  bytes
BANKW_       WR/     Text      13:55:41 31-Jul-2011  887  bytes
BANKW_T_     WR/     Text      17:40:51 07-Jan-2011 1709  bytes
CSVW_        WR/     Text      16:26:56 07-Nov-2010 3149  bytes
DATEW_       WR/     Text      21:11:50 21-Jul-2010 1409  bytes
DATEW_T_     WR/     Text      16:53:56 07-Mar-2010  987  bytes
FLAW_        WR/     Text      16:29:29 20-Mar-2011 1079  bytes
FLAW_T_      WR/     Text      16:51:10 19-Jun-2010 1341  bytes
PLOTW_       WR/     Text      12:28:44 01-Jan-2012 1078  bytes
STD_         WR/     Text      18:36:40 08-Jan-2011 1449  bytes
STORW_       WR/     Text      16:33:44 20-Mar-2011  880  bytes
STORW_T_     WR/     Text      13:24:53 04-Jun-2011 1096  bytes
STOW_        WR/     Text      16:32:00 20-Mar-2011  864  bytes
STOW_T_      WR/     Text      16:26:48 13-Aug-2011 2221  bytes
TRANSW_      WR/     Text      17:21:34 06-Nov-2011 2467  bytes
TRANSW_T_    WR/     Text      17:32:35 06-Nov-2011 2064  bytes
TRANSW_V_    WR/     Text      16:28:46 25-Jul-2010  945  bytes
TRANSW_V_T_  WR/     Text      15:16:12 02-Jul-2011 1241  bytes

*info !Konto.Resources.*
ResFind      WR/     BASIC     15:20:26 23-Aug-2006 1930  bytes
UK           D/      Directory 09:26:11 08-Oct-2011    0  bytes

*info !Konto.Resources.UK.*
Help         WR/WR   Text      10:19:15 21-Feb-2012 1461  bytes
Messages     WR/     Text      18:35:36 14-Dec-2011 1707  bytes
Res          WR/     Resource  10:18:36 21-Feb-2012   31 kbytes
```

```
*info !BMConvert.*
!Boot        WR/     Obey      10:51:36 02-Aug-2009   61  bytes
!Help        WR/     Obey      16:56:42 09-Oct-2011  183  bytes
!Run         WR/     Obey      12:44:32 09-Oct-2011  659  bytes
!RunImage    WR/R    ELF       18:24:45 12-Mar-2012  302 kbytes
!Sprites     WR/     Sprite    20:42:31 06-Aug-2009 2616  bytes
BMDLib       D/      Directory 17:21:22 08-Mar-2011   37 kbytes
cc           D/      Directory 01:56:29 02-Jan-1970 3037  bytes
Makefile     WR/     Makefile  15:08:43 02-Jul-2011  493  bytes
Makefile_amu WR/     Makefile  17:48:27 30-Dec-2011  504  bytes
o            D/      Directory 18:24:19 12-Mar-2012    0  bytes
Resources    D/      Directory 12:43:02 09-Oct-2011 1529  bytes
Settings     WR/R    Text      14:57:13 01-Jan-2012   40  bytes

*info !BMConvert.cc.*
!BMConvert   WR/     Text      18:55:19 07-Mar-2012   10 kbytes

*info !BMConvert.BMDLib.*
cc           D/      Directory 17:21:18 08-Mar-2011   16 kbytes
h            D/      Directory 17:21:22 08-Mar-2011   19 kbytes
Makefile     WR/     Makefile  18:38:40 13-Mar-2011 1995  bytes
Makefile_amu WR/     Makefile  17:31:32 30-Dec-2011    5 kbytes
o            D/      Directory 17:21:24 08-Mar-2011    0  bytes

*info !BMConvert.BMDLib.cc.*
ANALYS       WR/     Text      18:04:55 07-Mar-2012    6 kbytes
ANALYS_LST   WR/     Text      12:51:52 12-Jul-2009 1161  bytes
BANK         WR/     Text      08:33:09 03-Jun-2011    8 kbytes
BANK_LST     WR/     Text      15:43:56 06-Jul-2008 1145  bytes
BM           WR/     Text      16:42:34 11-Jun-2009    5 kbytes
ERR          WR/     Text      18:07:26 19-Nov-2008 3454  bytes
FILE         WR/     Text      15:39:38 27-Sep-2004 4068  bytes
LST_T        WR/     Text      18:09:10 25-Aug-2004 1357  bytes
SORT         WR/     Text      13:06:10 14-Sep-2008 1380  bytes
STO          WR/     Text      08:36:29 03-Jun-2011   12 kbytes
STO_LST      WR/     Text      12:43:22 12-Jul-2009 1148  bytes
TRANS        WR/     Text      17:11:52 04-Jun-2011   11 kbytes
TRANS_LST    WR/     Text      13:12:08 18-May-2008 1159  bytes

*info !BMConvert.Resources.*
ResFind      WR/     BASIC     15:20:26 23-Aug-2006 1930  bytes
UK           D/      Directory 12:43:14 09-Oct-2011  598  bytes

*info !BMConvert.Resources.UK.*
Help         WR/WR   Text      17:18:27 03-Dec-2011  186  bytes
Messages     WR/     Text      10:54:06 02-Aug-2009   63  bytes
Res          WR/     Resource  16:51:40 02-Aug-2009  940  bytes
```