Open Table InSpector

Database Browser

&

Library

for HMG Extended (minigui)

© Hans Marc

Belgium

2020…2021

(First release document : 03-08-2020)
(Last update document : 05-03-2021)

(Written while the world was fighting the Corona virus.)

Status :
FreeWare

Notice :
This program may not be sold or resold, distributed as a part any commercial package, used in a commercial environment, used or distributed in support of a commercial service, or used or distributed to support any kind of profit-generating activity, even if it is being distributed freely. If you would like to distribute this program as part of a commercial distribution, magazine, Internet book, CD ROM, etc. please contact us for permission.

Disclaimer :
This program is FreeWare and is provided "AS IS" without warranty of any kind, either express or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. In no event shall the author be liable for any damages whatsoever including direct, indirect, incidental, consequential, loss of business profits or special damages, even if the author has been advised of the possibility of such damages.

Contents

1. What is Otis.
2. What can you do with Otis.
3. How to use Otis.
4. Main screen.
5. Table Inspector screen.
6. Plugin mode.
7. LetoDbf.
8. Settings.
9. Command line options.
10. Files used & created by Otis.
11. Some screen previews.

What is OTIS

Otis stands for '**O**pen **T**able **I**n**S**pector' and is a database table inspection tool for Dbase, Clipper, Harbour, xHarbour, FoxPro, ..., files.

Otis has a integrated **DATASET manager**. This need some explanations.

A **DATASET** is a set of ONE or MORE tables and index files. You can save this SET of files with a name and load it back again later on with only one click. You can anytime you want add or delete tables and index files to a existing dataset and save it again with the same or another name.

The dataset manager permits you to inspect tables when you are developing and testing a program without cycling through the classic operations like :

- open table, open index file, inspect, modify, update, close
- open another table, open index file, inspect, close
- ...etc.

Create simply a **dataset** with tables and index files that you want to inspect on a regular base, save it, and reload all tables and index files with only one click. I can assure you, you will save a lot of time and many mouse clicks.

Why did i create this tool (or rather rewrote)

This is in reality a implementation of a old clipper tool that i wrote, i think, more than 20 years ago. But with far fewer possibilities as this new tool that i called Otis. We integrated this module (the old tool) in all our programs and protect it, if necessary, with a password.

Otis() helped me a lot of times for many many years already because it permits me, on site, to inspect, modify or repair data in a table at runtime on the fly in a simple and fast way. This without the necessity to install other external dbf viewer programs.

In 'Plugin mode', you can use it also for runtime debugging inspection of a table. Thus see 'almost' life updates from the running program. No 'modal' windows are used. By using this method, it permits you to switch between the running program and this tool when you want. You can even open multiple tables at the same time. Each table has its own dbfviewer, called 'Inspector' and all tools. Special precautions are taken for this mode to prevent data corruption when you try to modify a table at the same time as the running program. More details for the 'Plugin mode' in another chapter.

You ask me "Why rewrite yet another dbfviewer" ?
My answer, i found that there was always a thing missing in all other, very known, existing programs like dbu, dba, dbfview, mgdbu, ...etc. So i had to use one or more of them to view and or manipulate a dbf file. I don't want to say that those programs were bad, rather the contrary. I used them often, but i tried to regroup and integrate all possibilities of all those separated tools in a single program. It is certainly not perfect for everybody but i am open for suggestions.

I borrowed some code from other dbfviewer programs that are in the sample folders of hmgextended. All concerned parts of the program contains a remark and reference to the original source and author.

Another reason.
I wanted to update / upgrade the visual design to a new level, like the win10 flat design. I don't like toolbars with a lot of pull-down menus. You have to click, click and click endless everywhere to discover what is possible. My opinion is that a userinterface should be clear and eye-catching. At first look all "bells and whistles" should be visible and my experience (>30 years) showed me that the first user experience is very important for a program to be succesfull.

And not to forget, i could always count on the clipper harbour community if i had a question.

So this is my way to contribute. It would be a pleasure if you do give me some feedback.

ENJOY, i hope OTIS is of any use for you. Let me know ...

What can you do with Otis :

Support all Codepages known in Harbour.

Support for the following RDD drivers :

DBFCDX

DBFNTX

DBFNSX

SIXCDX

LETODBF

You can use a mix of rdd drivers in the same dataset.

Dataset manager (main screen) :

- Open tables, multiselect with index AutoOpen (cdx) support.
- Attach index files to a table, multiselect support.
- Save multiple tables and index files in a Dataset.
- Load a Dataset.

Table viewer tool called the 'Inspector' :

- Select a index / order. (Other options see index manager below.)
- Set a filter.
- Show / Hide deleted() records.
- Set / Clear a filelock.
- Lock (freeze) columns on screen.
- Show / Hide columns.
- Search and replace data, file wide or fields only, with **SCOPE**, **FOR** and **WHILE** expressions.
- Seek wizard :
  Presents a form with all fields used in the index KEY expression and autofills the seek expression.
  Seek first, Seek last and Set Exact on/off.
  Copy the 'seek expression' to the 'filter expression' textbox so that you can use the same expression to filter a table. Example, seek the first record and then show only records with the same field contents.
- Copy / paste a record :
  Paste a record to another record in the same table or
  Paste to another table.
  All fields or only a selection of fields in function of there visibility.
- Clear a record.
- Duplicate a record.
  Otis keeps into account the visibility of the columns/fields.
- Add / Insert records, one or more records at once.
- Up / Down, moves a record physically.
- Delete / Recall records with **SCOPE**, **FOR** and **WHILE** expressions.
- Pack / Zap a table.
- Append a file.
- Save a table to another table with the possibility to create a sub table.
  When a table is saved Otis takes into account active filters, index/orders and saves only visible fields. This permits to create a sub table with only the fields and data that you want.

Index manager :

- Create a new index, single or compound index files.
- Delete a index, tag.
- Reindex all orders.
- View detailed index info with the possibility to copy this info to the clipboard.

Table and index properties viewer :

- Export structure to a .csv file.
- Export structure to a .prg file. This prg can be used to create a table and all order index files.
- Export structure to the clipboard.
- View table info : filename, date, reccount, used rdd, used codepage, ...etc.
- View index info : list of all index files, tags, filenames, **KEY**, **FOR** and **WHILE** expressions.

Table structure editor :

- Create a new structure.
- Modify a existing structure.
- Import a existing table structure to create a new table.

3. How to use Otis :

You can use it in two different ways :

1. **Compile it as a STANDALONE executable.**
   (The executable of Otis is available for download.)
   The most simple method to use this program. Start Otis and add tables, indexes, load and save datasets, inspect tables, …etc.

2. **As a PLUGIN in your project.**
   That's where the magic happens.
   Include OTIS.PRG or OTIS.LIB in your project.

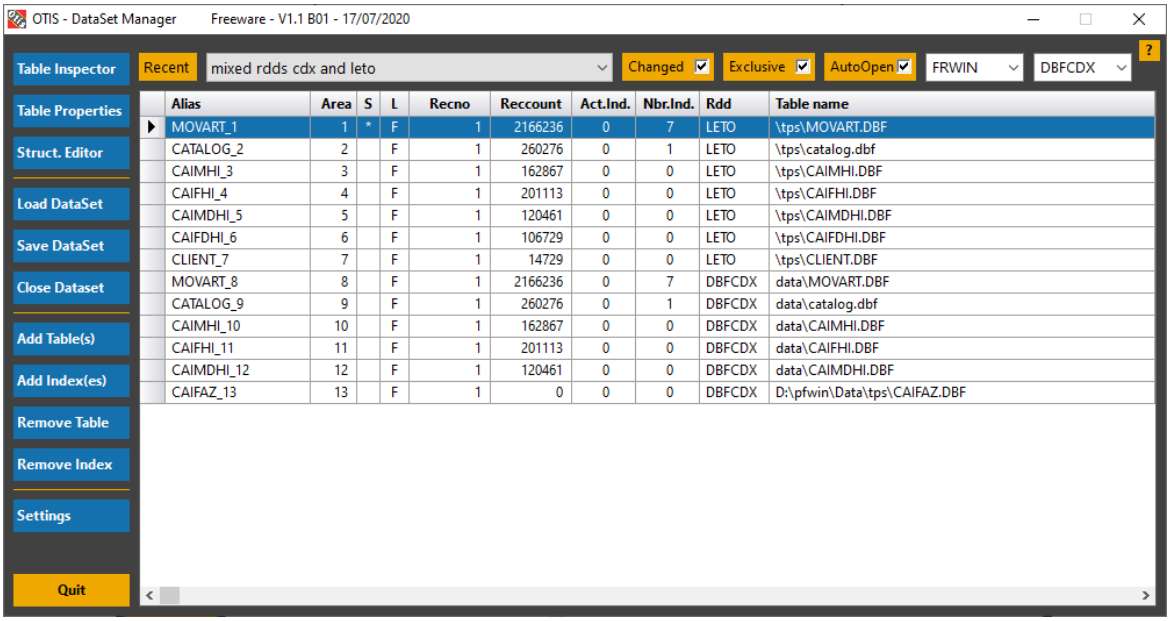
   First thing to do, attach function **Otis()** to a hotkey or assign it to, by example, a control "ACTION" property.

   When you call this function, Otis() scans **'ALL CURRENTLY OPEN TABLES AND INDEX FILES'** opened by your program and creates and present them in a new DATASET.

   You can save this dataset. Later on, when you reload this dataset in the Otis standalone version. You will find the same situation as if you where in your program to inspect and or modify data.

   You will find more specific information on how to use this mode below in another chapter.

4. Main screen of Otis



Alias	Area	S	L	Recno	Reccount	Act.Ind.	Nbr.Ind.	Rdd	Table name
MOVART_1	1	*	F	1	2166236	0	7	LETO	\tps\MOVART.DBF
CATALOG_2	2		F	1	260276	0	1	LETO	\tps\catalog.dbf
CAIMHI_3	3		F	1	162867	0	0	LETO	\tps\CAIMHI.DBF
CAIFHI_4	4		F	1	201113	0	0	LETO	\tps\CAIFHI.DBF
CAIMDHI_5	5		F	1	120461	0	0	LETO	\tps\CAIMDHI.DBF
CAIFDHI_6	6		F	1	106729	0	0	LETO	\tps\CAIFDHI.DBF
CLIENT_7	7		F	1	14729	0	0	LETO	\tps\CLIENT.DBF
MOVART_8	8		F	1	2166236	0	7	DBFCDX	data\MOVART.DBF
CATALOG_9	9		F	1	260276	0	1	DBFCDX	data\catalog.dbf
CAIMHI_10	10		F	1	162867	0	0	DBFCDX	data\CAIMHI.DBF
CAIFHI_11	11		F	1	201113	0	0	DBFCDX	data\CAIFHI.DBF
CAIMDHI_12	12		F	1	120461	0	0	DBFCDX	data\CAIMDHI.DBF
CAIFAZ_13	13		F	1	0	0	0	DBFCDX	D:\pfwin\Data\tps\CAIFAZ.DBF

Left side menus :

- **Table Inspector**
  Select one of the tables and click on this menu or double click a entry to open a table.
  You can open multiple 'Inspectors' at the same time.
- **Table Properties**
  Opens a window with the table structure information and menus to export the structure to a prg, a csv file or copy the structure to the clipboard for documentations purposes. The generated prg file can immediately be used in your own program to create a empty dbf file, index files and there orders.

- **Structure Editor**
  Create a new table or modify a imported existing structure.
- **Load, Save, Close Dataset**
  Does not need further explanation.
- **Add, Remove Table / Index,** idem.
- **Settings,** global program settings.

Top line option menus :

- **Recent**
  5 most recent used datasets.
- **Changed**
  Read only flag to indicate if a loaded dataset has been modified or created from empty.
- **Exclusive**
  Set this checkbox if you want to load tables in exclusive mode. This is necessary for some functions like, pack, zap, reindex, …etc. Otis refuse those operations and warns you if a table is not opened exclusive.
- **AutoOpen**
  Open index files automaticaly if it has the same name as the table. Not supported by the NTX rdd.
- **Codepage**
  Sets the default codepage that you want to use to open the tables. All existing codepages in Harbour are loaded in Otis.
- **Default Rdd**
  Sets the default rdd driver to use to open tables. You can change this setting when you want. Thus you can load one or more DBFNTX table, then load one or more DBFCDX tables and so on. The rdd used to open a table is saved in the dataset. You can thus mix rdd drivers in one dataset. When you reload a DATASET the proper rdd driver is used for each table.
  Supported drivers are :
  - DBFCXD
  - DBFNTX
  - DBFNSX
  - SIXCDX
  - **LETODBF**

Table browse control :

- **Alias name**
  Is the same as the table name but suffixed with the area number.
  Example, the alias for a table with filename ‘MOVART’ becomes ‘MOVART_1’.
- **Area number,**
  - In Standalone mode assigned automatically.
  - In Plugin mode determind by the running program. Remember we scan all areas from 1…65564.
- **S** of ‘selected’
  Filled with a ‘*’ if this table is the current ‘selected’ table.
- **L** of ‘lock’
  Three possibilities :
  - ‘F’ the file is locked, by a filelock or because the option ‘Exclusive’ was checked before load.
  - ‘R’ if there is a only one record locked.
  - ‘R+’ if multiple records are locked.
- **Recno**
  Current record number.
- **Reccount**
  Total number of records, unfiltered by a index,set filter or deleted() status.

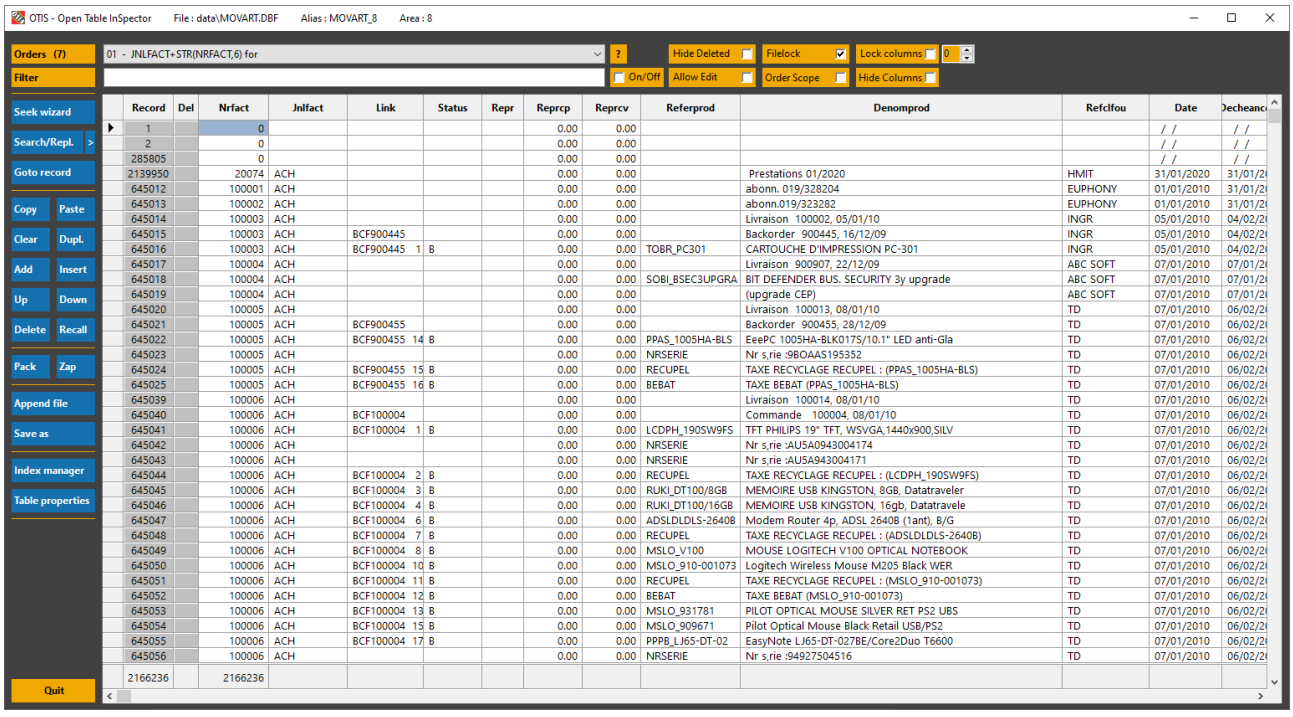
- **Act.Ind.**
  Actif order number.
- **Nbr.Ind.**
  Number of orders (not necessary the number of orderbag files, remember compound cdx files.)
- **Rdd**
  Rdd driver used to open the file.
- **Table name**
  three possibilites :
  - If LetoDbf Rdd, full data pathname of server.

  For others Rdd drivers :

  - Only the sub path name is saved with the filename if you open a table that is in a sub-folder of the current Otis.exe location. Thus without the drive letter or upper level paths. With this method you can move folder\tables to another location. Otis can reload them from the same location as long as they stay a subfolder of the running program.
  - Full pathname, drive, path and filename if a table is loaded in another folder and or disk.

5. Table Inspector screen :

Select one of the tables and click menu button 'Table Inspector' or simply double click a entry.
As already mentionned, you can open multiple tables each with there own 'Inspector' screen.



This is the real table tool center.

Top menu line :

- **Order (x) and combobox**
  Display the number of loaded orders for the table. Use the combobox to select one of the orders.
- **Filter**
  Filter expression that you can enable/disable with the checkbox just to the right of it.

- **Hide deleted**
  Enable/disable display of 'Deleted' records.

THIS IS A GLOBAL PROGRAM WIDE SETTING IN HARBOUR.
BE VERY CAREFULL TO USE THIS OPTION ESPECIALLY IN PLUGIN MODE.
IF YOU CHANGE THIS SETTING YOU CHANGE IT FOR ALL OPENED TABLES AT THE SAME TIME.

- **Allow Edit**
  Set this checkbox if you want to edit cells with a double click or toggle delete status of a record with the 'delete' key.
- **Filelock**
  Set a filelock if you want to prevent that other parts of a program can make changes. It is set automatically if you loaded the table exclusive and in consequence can not be unchecked.
- **Order scope**
  Opens a window where you can set a order TOP and BOTTOM scope. The checkbox permits to switch between scoped and unscoped display of a table.
  Each order can have its own scope.
- **Lock columns**
  Spinner and checkbox to freeze 'x' columns.
- **Hide columns**
  Permits to select the columns to be displayed. Use the checkbox to switch between the two modes.

Left side menus :

- **Seek wizard**
  opens a window wherein you can define a seek expression. The wizard creates a form with all the fields used in the expression. You only need to fill in those fields and the wizard will automatically create the full expression in function of the used index key. This wizard is not perfect and some complex key expressions can not (yet) be treated by the wizard. In that case disable the autofill function and enter the expression manually.
  You can seek the first record, the last record, set exact on/off and copy the expression to the filter.
- **Search / Repl.**
  Two methods.
  The first one to search and / or replace data in a table for only one fields or for all fields at the same time.
  The second one to replace data in only one field at the time.
  Replace field contents with a '**Value'** or with the help of a '**Expression'**.
  Options are available to **set a scope** and set a **FOR** and / or a **WHILE** expression if needed.
- **Goto record**
  Goto top, bottom or goto a specific record.
  Don't forget that you can also use ctrl-Home/End to go to the top or bottom of a table.
  It is obvious that this is always in function of a actif index and or actif filter.
- **Copy, Paste**
  The copie / paste buffer is a **global** program buffer.
  You can thus copy/paste contents from **one record to another record in the same table** or to **another table**.

  Otis even goes further then what is usually possible.
  Sometimes it is necessary to copy/paste only some specific fields and not the whole record.
  Otis makes it possible. How to proceed. Hide the columns (fields) in the 'Inspector' that contains the database where you want to copy from. Click, copy, only the visible fiields are copied into the copy/paste buffer. Switch to full columns view, goto another record and click paste. Again, only the previous visible columns (fields) are pasted.

And as already mentionned, you can even copy fields from one table to another table. Remember, the copy/paste buffer is a global program buffer. How to proceed. The same as if you copy between records in the same table but before clicking the paste tool, choose another table. Fieldnames that do not exist in the target table or are not of the same type are ignored. Again, only visible columns (fields) are copied / pasted.

- **Clear a record**
  Does not need explanations.
- **Duplicate a record**
  The selected record is pushed down.
  Otis duplicates the record but only duplicates data of the visible columns (fields).
- **Add, Insert**
  You can add / insert multiple records in one run.
  Insert needs some special attention.
  Insert is only possible if the table is opened in **exclusive** mode. We use the recently introduced new low level function **DBINSERT()** that is very very fast. The only inconvenience is that dbinsert() ignores actif orders. Otis recreates all orders after it.
- **Up, Down**
  Physically moves a record up or down.
  One limitation, this function can only be used if there is no order actif.
- **Delete, Recall**
  You can delete / recall a single or multiple records, set a **SCOPE** and if necessary, set a **FOR** and/or **WHILE** expression.
  You can also use the 'delete' key to delete/recall a single record if the 'Allow edit' box is checked.
- **Pack, Zap**
  Does not need explanations only that the table must be opened in **exclusive** mode.
- **Append file**
  Does not need explanations.
- **Save as**
  Needs some special attention.
  If you save a file only the records and columns (fields) that are visible in the browse table are saved.
  Options like a filter, hidden columns, scope, index, set deleted, they all defines what is saved.
  You can thus easily create a subset of a existing table.
- **Index wizard**
  Order management wizard.
  Permits to create, delete, modify order tags and files.
  Reindex all orders for a table.
  The filename of the index file is automatically proposed by Otis if you create a new order. Sure, you can modify the proposed name.
  Single order (ntx) and multi-tag, compound cdx files are supported.
- **Table properties**
  Show the structure of the table.
  Menus available :
  - Export the structure to a **prg** file that you can use in your own program to create a new empty table and attached index files.
  - Export structure to a **csv** file.
  - Export structure to the windows clipboard.
  - Table info, show all kind of table info and a menu to copy this info to the clipboard.
  - Index info, show index info, filenames, key, for and while clause and a menu to copy this info to the clipboard.

I think that almost all tools are there ….
If you miss one, let me know ….

6. Plugin mode

As explained you can include Otis in your own program by including the LIB file in your project.

Attach the function **Otis()** to a event like a special key or a control action event. No other functions have to be called to initialize, set, reset or whatever else.

Otis knows when it is in 'plugin mode' or running as a 'standalone' executable. How does Otis know that it is running in 'Standalone' mode. Simply by checking if the word 'OTIS' is found in the executable filename. You can thus create your own executable with your own special SET(….)tings as long as the executable name contains 'OTIS'.

Differences between 'Standalone' and 'Plugin mode' ?

In Plugin mode Otis inherits from your application :

- All your program settings SET(…).
- Codepages.
- Rdd drivers.
- Deleted status.
- Exclusive open mode.
- AutoOpen mode.
- All Alias() names.
- All area numbers.
- All table record positions are retrieved.
- All table 'Filter' settings are retrieved.

The corresponding checkbox and or combobox in the main screen reflects these current settings.

There is one thing that needs special attention and it is a very, very important issue in 'plugin' mode :

When you call OTIS() it scan all areas for tables and indexes opened by your program and present them in a new dataset. You can open a table in the 'Inspector' window and this is where big problems can arise. You will say, why.

An example will speak for itself.

We are running under windows and as you know all forms can be clicked to give them the focus. Remember there are only a few 'Modal' forms in Otis. You can thus, at any moment, set the focus to a form in your application and then give focus to a form in Otis. What happens if a table is opened in a form in your program and the record pointer is positioned on a specific record. Remember, this table is also shown in the dataset with the consequence that you can open this table in the 'Inspector' at the same time. When you open this table, the 'Inspector' it will be positioned on the same record. What happens if you, in Otis, change the record position, delete / add a record, change order, set a filter, set a scope ? Your application form completely loose control in this table because Otis is working in the same area. This is a dangerous situation and data integrity is not assured. Data could be corrupted or could be lost.

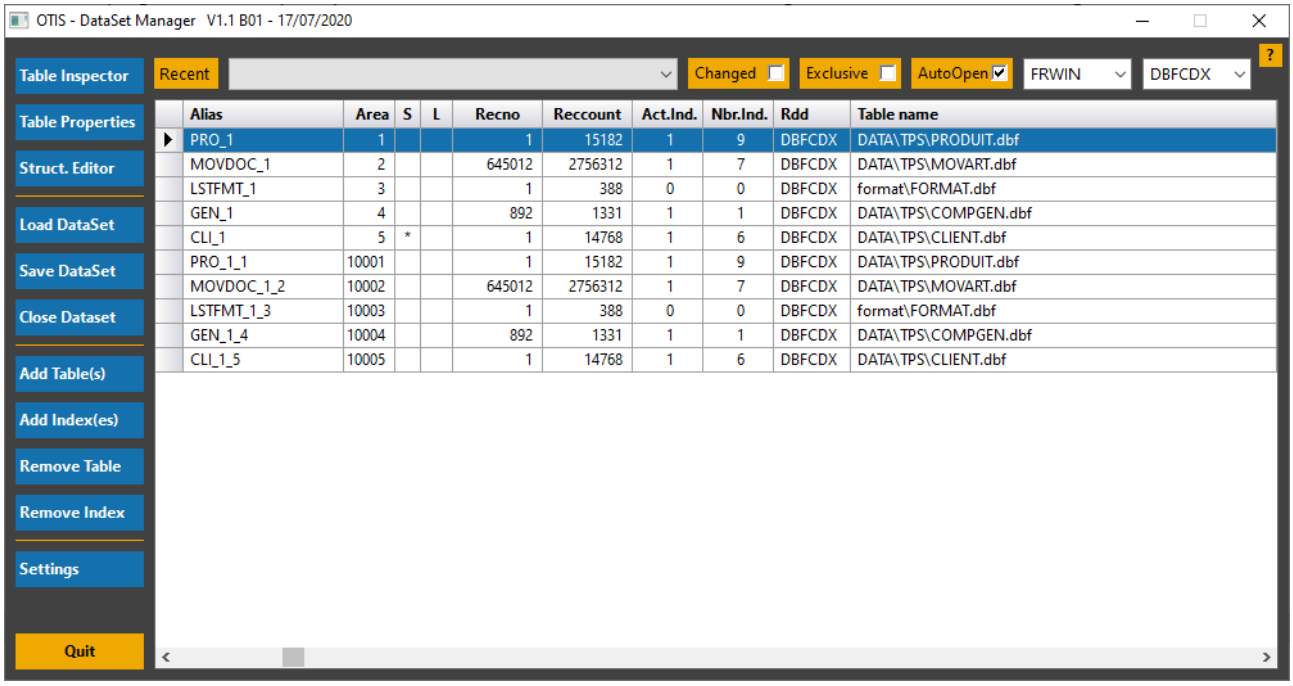
I have found the following solution to prevent these situations :
As said when you call Otis() it scans all areas. Opened tables and indexes are presented in the dataset table with the alias and area number set by your program. I call this area the 'mother' area. What Otis do is reopen the same table with another alias name and another area number (see Settings menu). The default area number for this reopened table is the 'mother' area number plus a offset of 10000. You will see thus each table in what i call a 'Ghost' table with his own alias and area number and record positioning.

We advise you to open a table opened by your application in this 'Ghost' area because almost all operations like, change order, filter, go to another record, …etc. are independent of the 'mother' area. Your program form that is positioned on a record will not be influenced by operations in this 'Ghost' area.

Example below :

'**Mother**' table name is produit.dbf :
The alias name is 'Pro_1' and the area number is '1'.

'**Ghost**' table :
Alias will be 'Pro_1_1' with area number '10001'.



Alias	Area	S	L	Recno	Reccount	Act.Ind.	Nbr.Ind.	Rdd	Table name
PRO_1	1			1	15182	1	9	DBFCDX	DATA\TPS\PRODUIT.dbf
MOVDOC_1	2			645012	2756312	1	7	DBFCDX	DATA\TPS\MOVART.dbf
LSTFMT_1	3			1	388	0	0	DBFCDX	format\FORMAT.dbf
GEN_1	4			892	1331	1	1	DBFCDX	DATA\TPS\COMPGEN.dbf
CLI_1	5	*		1	14768	1	6	DBFCDX	DATA\TPS\CLIENT.dbf
PRO_1_1	10001			1	15182	1	9	DBFCDX	DATA\TPS\PRODUIT.dbf
MOVDOC_1_2	10002			645012	2756312	1	7	DBFCDX	DATA\TPS\MOVART.dbf
LSTFMT_1_3	10003			1	388	0	0	DBFCDX	format\FORMAT.dbf
GEN_1_4	10004			892	1331	1	1	DBFCDX	DATA\TPS\COMPGEN.dbf
CLI_1_5	10005			1	14768	1	6	DBFCDX	DATA\TPS\CLIENT.dbf

Be carefull, even with this solution, some tools in Otis can have an impact in the 'mother' area and result in data loss.

This is maybe not the best solution but the only one that came up for the moment.

Last remark, when you are in 'plugin' mode and close a dataset, only the 'Ghost' areas are closed. Otis may not close tables opened by the running program. The difference is made by the area number. All areas with a area number greater than >= 10000 are closed because they are 'Ghost' areas.

7. Letodbf

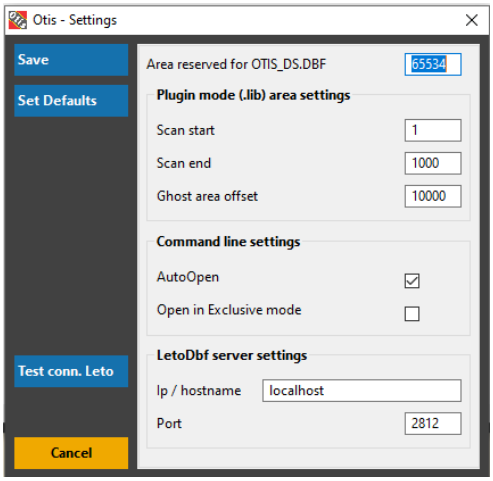
Otis contains all the necessary drivers to open and manage LetoDbf tables.
LetoDbf is really a excellent client server driver. Thanks to Elch.

The only thing that you have to do is set the servername and port to use for a connexion in the 'Settings' menu.

There is no 'connect' button or whatever else. Otis try to open automatically the connexion when it is not yet established when you load a previous saved dataset that contains tables that use the LetoDbf driver or when you want to add a table to a dataset with the LetoDbf rdd driver selected in the combobox. Otis contains special getfile() and putfile() functions to work with the LetoDbf server '**Data'** folder (see letodb.ini).

8. Program settings

There are few program settings.



- **Area reserved for OTIS_DS.DBF**
  Area number exclusively used for the program dataset table, OTIS_DS.DBF.
  Change it if there is a conflict with a area number used by your program when you run in 'plugin' mode.
  Attention (2^16) – 1 = 65535. Harbour excludes this area number, thats why i used by default 65534.

- **Plugin mode area settings :**
  Start and end area range scanned when Otis() is called to fill the dataset with the opened tables by your program.
  Ghost area offset, offset added to the 'mother' area number for the 'ghost' areas.

- **Command line mode setttings :**
  AutoOpen mode.
  Exclusive open mode.

- **LetoDbf settings**
  Ip / Hostname of the LetoDbf server.
  Can be :
  ip address, 192.168.1.253
  a hostname LETODBF.DDNS.NET
  localhost
  …etc.

Port, port number to use.
Don't forget for your firewall that LetoDbf use 2 ports. The one that you specify here and the next one.
Example, 2812 and 2813.

9. CLI, Command Line Options :

Otis table1.dbf, table2.db, …etc.

Filename separator can be ',' or ' ;'.

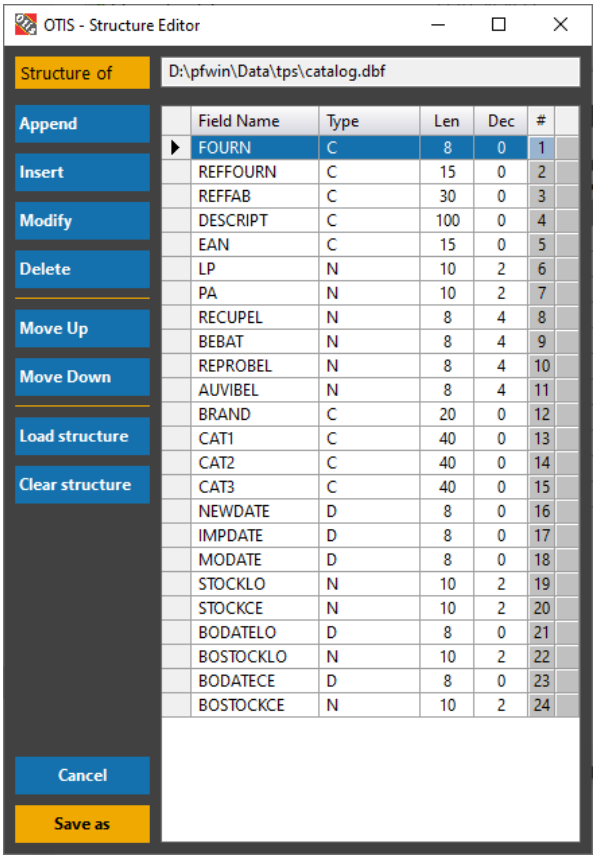
10. Files used and created by Otis :

Otis.exe or Otis.lib creates automatically the following files if they are not found in the folder from where you launch the program :

Otis.ini	: contains all program settings.
Otis_ds.dbf	: database for dataset storage.

Copy these two files, if you copy Otis.exe from one folder to another folder, if you want to keep the same settings.
I choose for this solution because in 'plugin ' mode with the lib file some settings, like by example the Letdbf server settings, could be different per application.

11. Some previews :



Structure Editor.

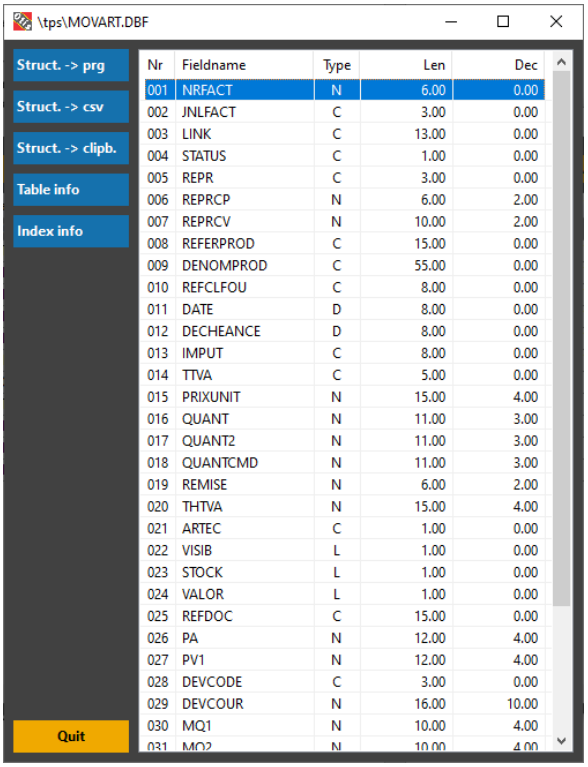


Table properties.

```
//
// This code is generated by OTIS()
//
// Otis stands for 'Open Table InSpector'.
//
// Table   ; data\CLIENT.DBF
// Date    ; 20/01/2021
// Time    ; 22:01:31
//
// Index info :
//       Orderbags   : 1
//       Orders      : 2
//
//   1  OrderBag name   : CLIENT.CDX
//          Order nbr   : 1
//          Order name  : REFER
//          Key         : REFER
//          For         : !DELETED()
//
//   2  OrderBag name   : CLIENT.CDX
//          Order nbr   : 2
//          Order name  : NOM1
//          Key         : NOM1
//          For         :
//
Function Dbf_Create_CLIENT()

  DBCREATE( 'CLIENT', {;
          { 'REFER'     , 'C',   8,   0 } ,;
          { 'NOM1'      , 'C',  30,   0 } ,;
          { 'NOM2'      , 'C',  30,   0 } ,;
          { 'ADRESS'    , 'C',  30,   0 } ,;
          { 'SENDTO'    , 'C',   1,   0 } ,;
          { 'EMAILINV'  , 'C',  50,   0 } ;
          }, 'DBFCDX' )

   // Create index files
   USE CLIENT.DBF NEW

   INDEX ON  REFER ;
         TAG REFER ;
         TO  CLIENT ;
         FOR !DELETED()

   INDEX ON  UPPER(SUBSTR(NOM1,1,10)) ;
         TAG NOM1 ;
         TO  CLIENT
   USE

Return nil
```

Example ‘Export to prg’.

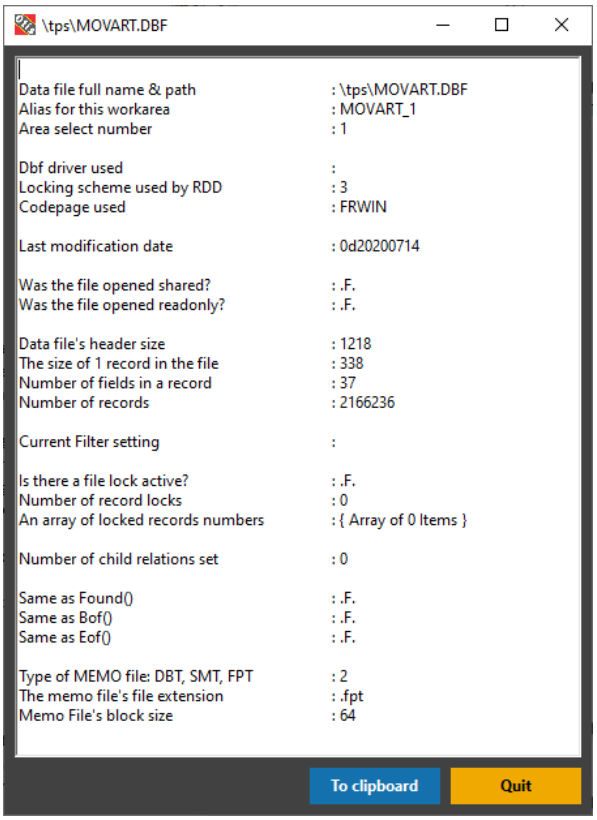
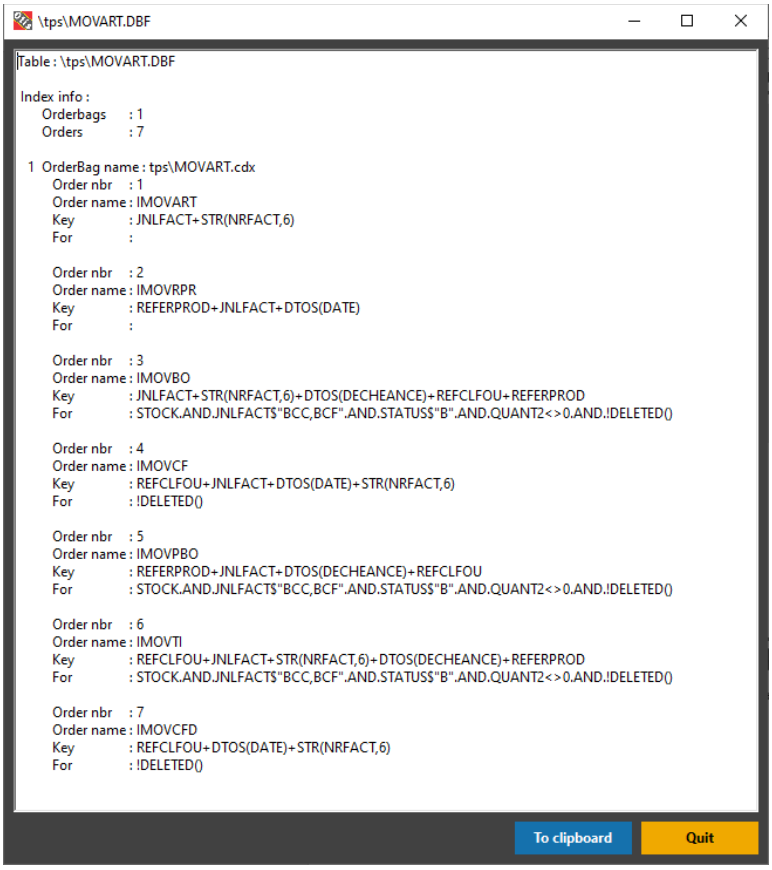
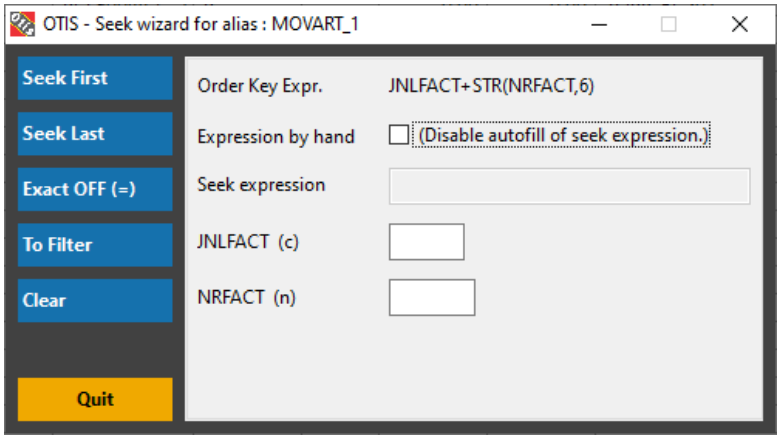


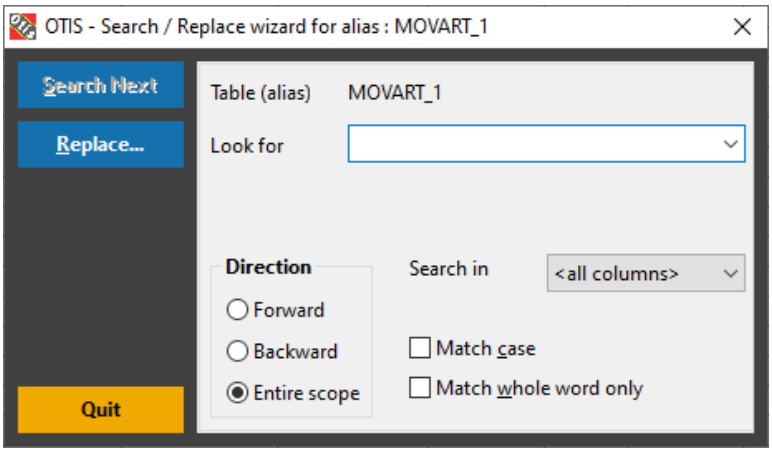
Table info.



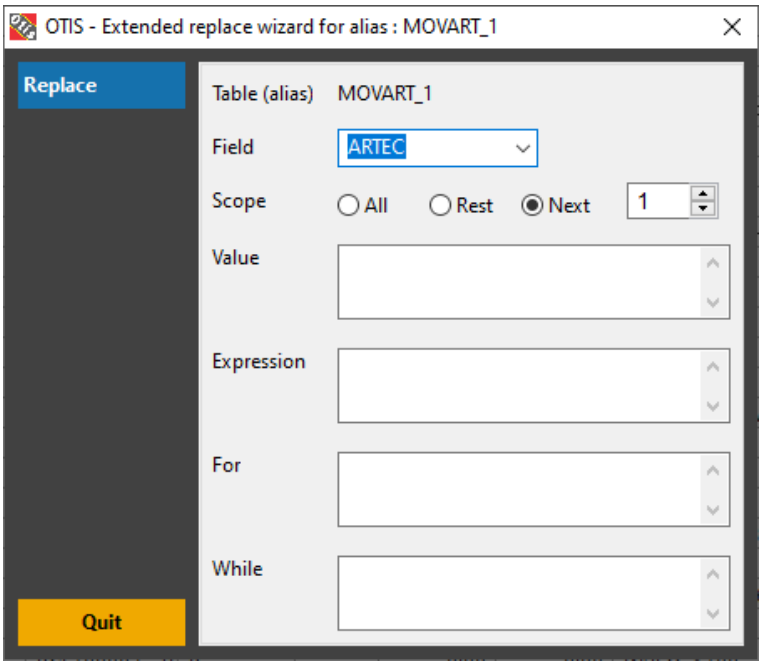
Index info.



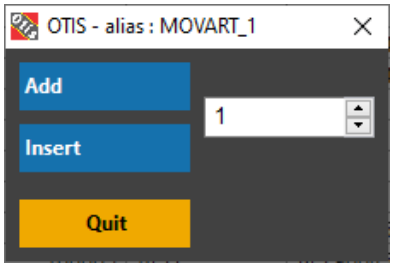
Seek wizard.



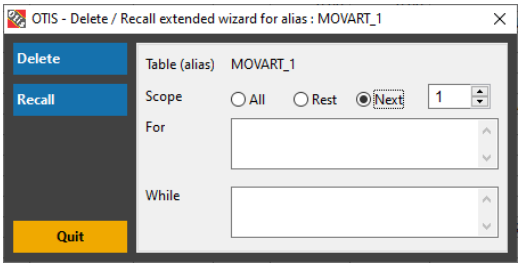
Search / Replace (standard version).



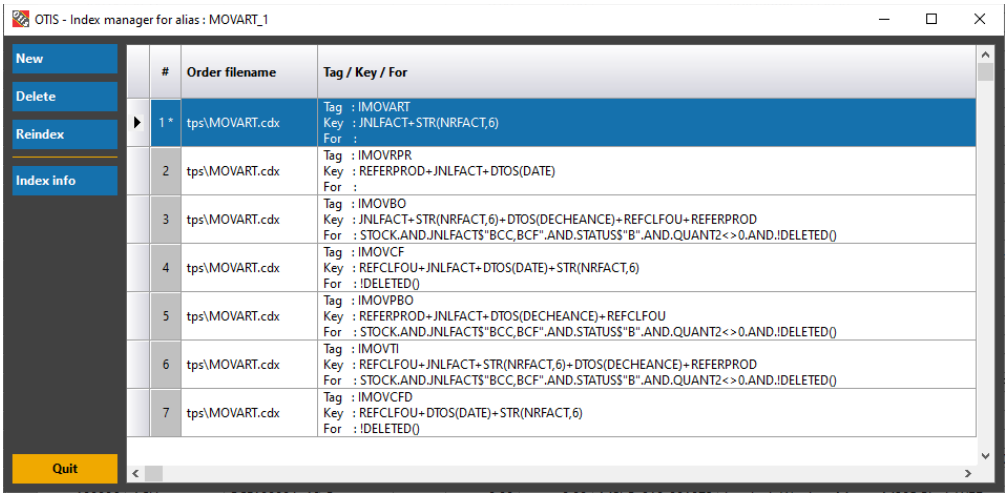
Replace wizard (extended version).



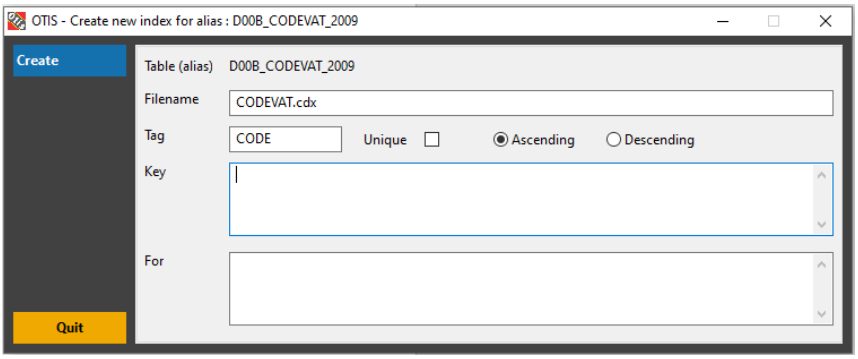
Add / Insert one or multiple record(s).



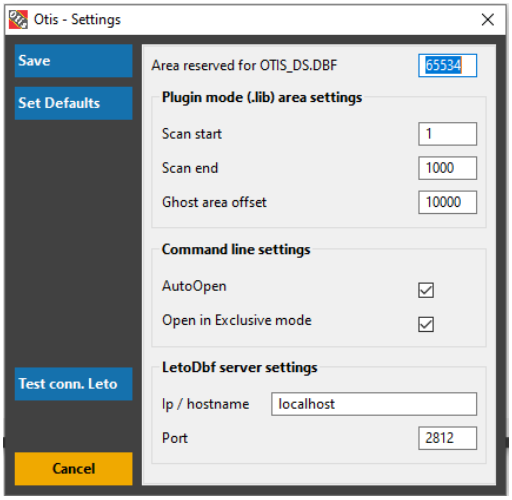
Delete / Recall one or multiple record(s).



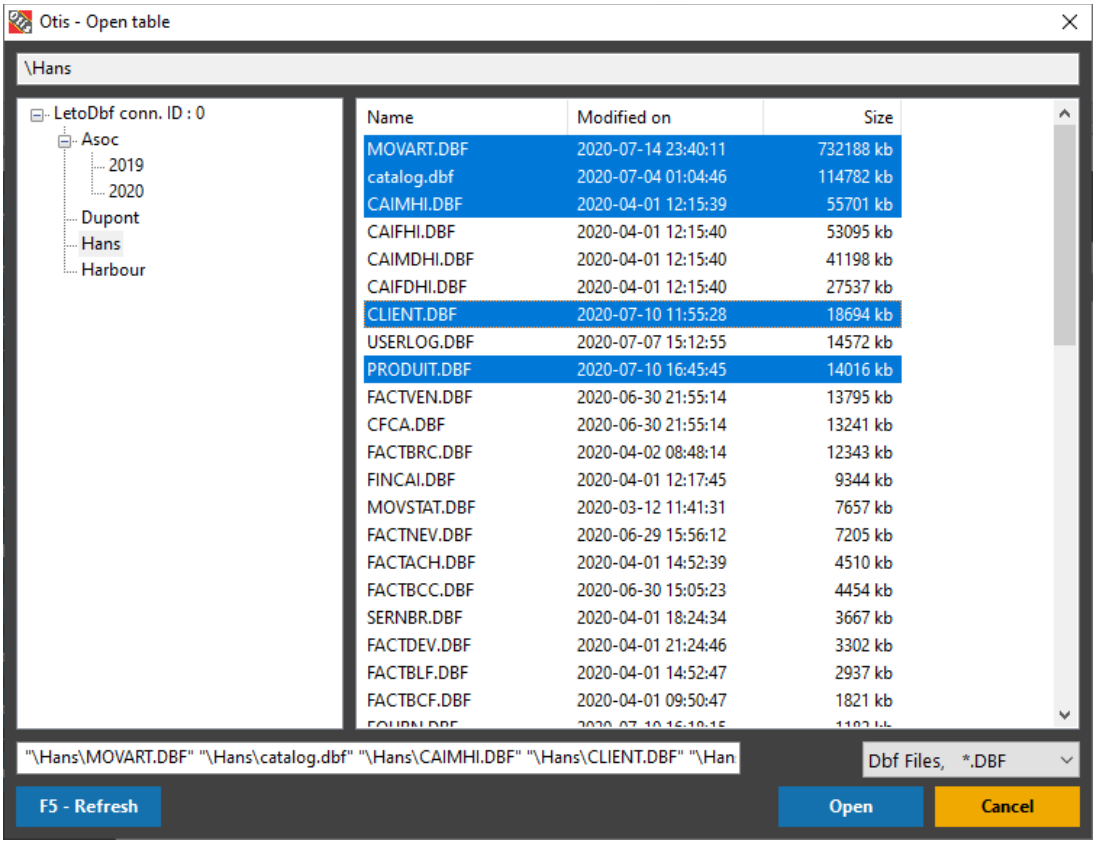
Index manager.



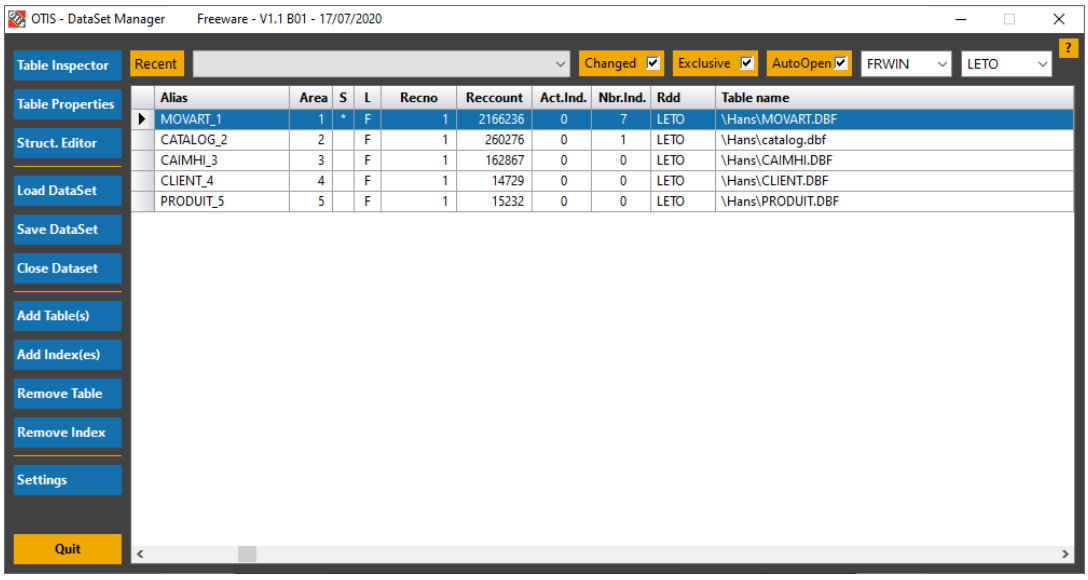
Create a new order.



Program Settings



Add one or multiple tables with rdd LetoDbf filepicker.



And resulting dataset in Otis.
Remark the first 2 tables, there order files are opened automatically.