Advanced Table Search Dynamic Link Library
Version 1·0

Designer   **Craig Ward (100554,2072)**
Date   **14/9/95**

Function   **(IMHO) an awesome advanced table search dialog box.**

****

My Terms

This product has been released into the public domain, **FREE!** I want no money in return, and you are welcome to make as many copies, and distribute them, as you like. **However,** I would appreciate some feedback, especially bugs or improvements that you think that I should make.

Use

This is a pretty funky dialog box. It allows you to search alphabetically through a table (via an alphabetically tabbed control), or numerically through a value specified by the user, or through an SQL query. It's title is "Advanced Table Search", which I think it is.

The dialog box contains:

[1] a combo box (which will populate automatically on creation with the fields from the table that you specified!)
[2] an edit-box for numeric search values, a DBGrid which will contain the answer set which is linked to an alphabetically tabbed tab-Set allowing quick searches\sorts
[3] an edit-box which the dialog box displays the actual SQL qeury it is running (allowing the user to amend and thus shoot their own queries at the data)

Calling the DLL

The DLL contanis one function which you need to call:

function ShowSearchDlg(sAlias,sHelp,sTable: **string**): **string**; **export**;

, where:
        sAlias = the alias\database name
        sHelp = the path and filename of the calling application's helpfile (**with extension**)
        sTable= the table to be used in the search dialog (**no extension!!!!!**)

Note: the dialog box has a hard-coded help context of **102 ,** so to provide context-sensitive help in your HLP file, ensure that there is a page for a context of 102.

Returned Values

If the user presses the OK button, then **the dialog box will return the value of the first field in the table, at the currently selected row**. Since I did not know what the field type would be, (ie: an integer, number, string and so on) I decided to compromise: **the DLL will return a string value**, which you can then convert in your host application if you need to search for an integer (in

Delphi this would be with the useful "**StrToInt**" function).

If the user presses the cancel button the DLL returns the string: "**UserCancel**" which you could provide some default handling of in your application.

For example (in **Delphi**):

```
{declaration in the calling unit - place after compiler directives}
function ShowSearchDlg(sAlias,sHelp,sTable: string): string; far;
external 'search';

procedure TForm1.Button1Click(Sender: TObject);
var
hDLL: THandle;
s: string;
begin
hDLL := LoadLibrary('search.dll'); {will work if search.dll is in the path}
if hDLL > HINSTANCE_ERROR then
 try
  s := ShowSearchDlg('DBDEMOS','c:\windows\progman.hlp','customer');
  if CompareStr(s,'UserCancel') = 0 then
   exit
  else
   begin
    try
     Table1.FindKey([ (StrToInt(s)) ]);
    except
     showMessage('Could not find key!');
    end;
   end;
 finally
  FreeLibrary(hDLL);
 end;
end;
```

For example (in **Paradox**):

```
uses SEARCH
ShowSearchDlg(sAlias CPTR, sHelp CPTR, sTable CPTR) CPTR
endUses

method pushButton(var eventInfo Event)
var
s                   String
endvar
s = ShowSearchDlg("myAlias","c:\\windows\\myHelp.hlp","myTable ")
myTab.locate("myField", s) ;where myTab is a table-frame
endmethod
```

Note: To be honest I did not have the time to get this to run in Paradox. Though there is no reason why it should not work since I can call it from Delphi with no problems. **There may be an error in the syntax above, which is likely since I don't use much DLL calling from Paradox** (infact I don't use Paradox much since I've got Delphi!!!). If anyone figures it out, please give me a call!!!!!!!!!!!!!

<u>Files</u>

The Search.DLL file can be where you like - just ensure that your calling application can reach it!!!

<u>Conclusions</u>

I hope that it helps. **Have fun and don't forget to give me some feedback!**

".....the truth is out there......"