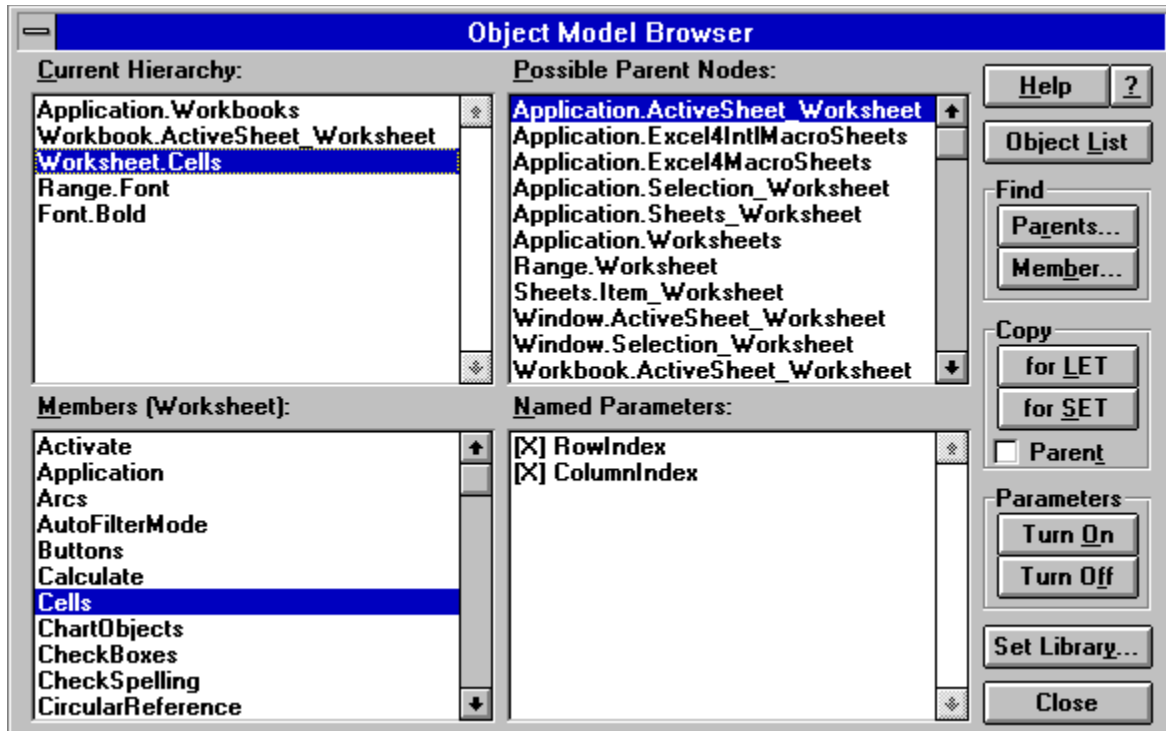This example of the **Object Model Browser Addin** (OMB.XLA) is based on the Microsoft Excel 5.0 object model. This dialog can be started through the **Visual Basic: View: Object Model Browser...** menu, the ⊞ **ToolbarButton** provided with the addin, or with **Ctrl-Shift-F2**. Click the areas of the graphic for a description of the dialog.

Object Model Browser

Current Hierarchy:
- Application.Workbooks
- Workbook.ActiveSheet_Worksheet
- **Worksheet.Cells**
- Range.Font
- Font.Bold

Possible Parent Nodes:
- **Application.ActiveSheet_Worksheet**
- Application.Excel4IntlMacroSheets
- Application.Excel4MacroSheets
- Application.Selection_Worksheet
- Application.Sheets_Worksheet
- Application.Worksheets
- Range.Worksheet
- Sheets.Item_Worksheet
- Window.ActiveSheet_Worksheet
- Window.Selection_Worksheet
- Workbook.ActiveSheet_Worksheet

Members (Worksheet):
- Activate
- Application
- Arcs
- AutoFilterMode
- Buttons
- Calculate
- **Cells**
- ChartObjects
- CheckBoxes
- CheckSpelling
- CircularReference

Named Parameters:
- [X] RowIndex
- [X] ColumnIndex

Buttons:
- Help ?
- Object List
- **Find**
  - Parents...
  - Member...
- **Copy**
  - for LET
  - for SET
  - ☐ Parent
- **Parameters**
  - Turn On
  - Turn Off
- Set Library...
- Close

Double click an item in the **Current Hierarchy** list to select the current position in the object hierarchy.   All lists will update to display the data associated with the new item.   Items in this list have the form <object type>.<member>.   For example, double clicking the Workbook.ActiveSheet_Worksheet item will change the <u>Members</u> list to show all methods and properties of the Workbook object, with ActiveSheet_Worksheet selected.   The underscore in the name implies that the ActiveSheet method can return more than one type of object. ActiveSheet_Chart will point to a Chart instead of a Worksheet object.   The <u>Parent Nodes</u> list will indicate all methods in all objects which return a Workbook object.   For example, Application.ActiveWorkbook and Application.ThisWorkbook will be in the Parent Nodes list. The state of the <u>Copy for LET</u> and <u>Copy for SET</u> buttons will change depending on the current selection of this list.

Double click an item in the **Possible Parent Nodes** list to change the parent of the current item in the hierarchy.   In the example, double clicking Window.ActiveSheet_Worksheet will replace Application.ActiveWorkbook and Workbook.ActiveSheet_Worksheet with Window.ActiveSheet_Worksheet.   In general, all information in the path leading to the current position in the hierarchy will be lost, and the newly selected parent node will become the first and current position in the hierarchy.   You can continue to work up one level at a time until no more parents are available.   The parent nodes list will only be empty when the Application object is at the top of the current hierarchy.

Double click an item in the **Members** list to change the current item in the hierarchy.   If the new item points to another object, the current position in the Current Hierarchy list will move down one and the Members list will update to display the properties and methods of the new object.   The Current Hierarchy will update recursively based on the last selected member of the new object.   For example, the next time the Font any method which returns a font object is selected in the methods list, the hierarchy will point to <object>.Font and Font.Bold.   If the current settings of the new object recurse, then the recursion stops and the Copy for LET button is disabled.   This happens with Range.Cells, which returns a Range object.   The object model browser will stop on the second Range.Cells line.   Choose another member of the Range object to continue.   The path through the hierarchy is valid when the LET button is reenabled.
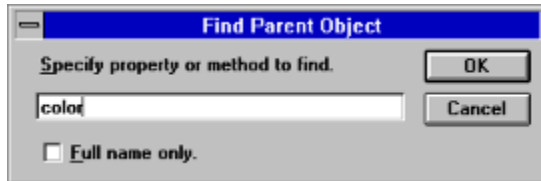
The **Named Parameters** list contains all of the named parameters for the current item in the hierarchy.   Optional parameters are indicated by the [ ] or [X] preceding the name of the parameter.   A checked box indicates the parameter will be copied when the VBA string for the current hierarchy is built.   If the named parameter is an optional Index, changing it will change the object pointed to by the current item in the hierarchy.   Application.Charts(Index) points to a Chart object, while Application.Charts points to the Charts collection.   Parameters in the list can be toggled by double clicking on them, or turned on and off the provided buttons.   The list is multiselect, so multiple parameters can be set at once.

The **Help** button shows the help topic for the currently selected item in the <u>Members</u> list.   If the Members list is empty, help is shown for the selected object in the <u>Objects List</u>.
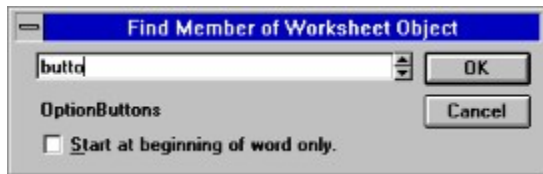
Show this help file.

The **Object List** button starts a new session by clearing all the lists and returning a list of all Objects in the Parent Nodes list.

Find Parents prompts the user to enter the full name of a method or property.   All objects are searched, and a list of all objects which have a corresponding member is placed in the <u>Possible Parent Nodes</u> list.   If only one matching object is found, then it is placed directly in the <u>Current Hierarchy</u> list.   If <u>Full name only</u> is not specified, a secondary list is shown with all possible properties and methods.   The search string must be at least three characters long.

Find Member will find a member of the current object, or an object in the Object List.   A search is executed for every key typed.   If Start at beginning of word only were checked, the search in the example would have found **Buttons** instead of **OptionButtons**.   Searching anywhere in the word is particularly useful for finding Constants, which generally start with the same two letters.

Copy for LET shows a message box with the VBA string corresponding to the current hierarchy. Choosing OK will copy the string to the clipboard to paste into VBA code. The example returns the string ***Workbooks(Index:=).ActiveSheet.Cells(RowIndex:=, ColumnIndex:=).Font.Bold*** to the clipboard. The Application object is only prepended when necessary. If the Application object is not the first object, the string will begin with a period, or the uppercased name of the object if the Parent box is checked. The uppercased parent name should be replaced in your VBA code.

Copy for SET returns a VBA string for use in a Set statement.   The object of the current position in the hierarchy is returned.   The example copies ***Workbooks(Index:=).ActiveSheet*** to the clipboard.   If the first item in the current hierarchy list is selected, the SET button is disabled.   The Application object is only prepended when necessary.   If the Application object is not the first object, the string will begin with a period, or the uppercased name of the object if the Parent box is checked.   The uppercased parent name should be replaced in your VBA code.

If the **Parent** box is checked, the uppercased name of the top object in the list is placed at the beginning of the copied string.   This name should be replaced in VBA code by an object of the given type.

The **Turn On** and **Turn Off** buttons are provided to set the state of selected objects in the Named Parameters list.   If a selected parameter is not optional, no change is made.   If no parameter is selected, the first parameter is set.

Set Library is used to determine the current set of data used by the object model browser. XLEN50.OMB is used in this example.   However, this works equally well with correctly formatted data from Microsoft Project or Excel object libraries in other languages.   The .OMB files should be placed in the directories of the corresponding applications so the help file can be found.   The current library is saved in the OMB.INI file and is automatically loaded with the Browser.   Any changes made through navigation of the object model are saved when the Browser closes or the current Library is switched.   If you do not want to save the changes, which indicate the last accessed property or method of each object, make the .OMB file read only.

Close the Object Model Browser dialog.