How-to Guide

Mobile Client Technology

How To… Use UnBound Columns in Grid

Version 1.00 – August 2007

Applicable Release: CRM Mobile Technology 4.0 and 5.0

1 Introduction

Sometimes it is required to show data from a data source in the list tile, other than the underlying business object (attached to the anchor supplying data to the grid).This can be achieved by modeling the grid with certain UnBound columns and populating the values for such columns at runtime using events raised by the grid control.

2 Implementation

2.1 Design Time

In the Mobile Application Studio (MAS), while the grid is being modeled, you would typically model an anchor with an underlying business object type. This would be mostly as a relation to another anchor, whose result would be a business collection. Each of the columns in the grid would be modeled with a particular attribute of the business object. Such columns are called Bound columns.

Along with these regular columns, you need to model some columns without any attribute. Such columns are called UnBound columns.

2.2 Runtime

At runtime, the grid works on the pull mode for showing data. That means that anytime the grid control is suppose to fetch new or updated data to be shown on the screen, it will ask for the data. This can also be forced on to the grid by calling a refresh on it (this discussion is out of the scope of this document).

Every time, the grid asks for data, it will be specific for a particular row. For example, if you are seeing rows 1 to 10 on the screen, then the grid may ask for data for each of the rows from 1 to 10. The grid control asks for data from the UI object *CoreCtrlGrid*. It is the responsibility of the *CoreCtrlGrid* to fetch proper data and send it to the grid control to be shown on the screen.
For all the columns, the *CoreCtrlGrid* will fetch data from the Business Anchor and will send it to the grid.

This logic will only work for the columns which have a business object attribute and will not work for UnBound columns. For UnBound columns, the *CoreCtrlGrid* will raise an event to the application code, which then will send the data for such UnBound columns.

In the application code for the list tile, you need to capture the following event from the grid control.

```
Public Event RowLoaded2(ByVal bk As Integer,
                        ByVal columns As SCRRUNLIB.Dictionary)
```

Where,
bk is the row number for which data is being asked for.
columns is the scripting dictionary of the values in all the columns for that row.

The above event is raised for each row that the grid asked data for.

Note:
The columns dictionary contains the values for each of the columns in a row, irrespective of whether it is UnBound or bound.

The application developer has now the option to enter values for the UnBound columns in this dictionary and send it back to the grid.

Note:
Values added for Bound columns will be ignored and only UnBound column values will be used by the *CoreCtrlGrid*.

2.3 Example

The application code has to be written as follows:

```
Private Sub ctrlMygrid_RowLoaded2(ByVal bk As Integer, ByVal
columns As Scripting.Dictionary) Handles ctrlMygrid.RowLoaded2

'The following values being set will be used, since the controls will be a UnBound columns
  columns.Item(ctrlUnBnd1.bookmark) =
                                       MyOwnDataSource.Item(“Value1”)
  columns.Item(ctrlUnBnd2.bookmark) =
                                       MyOwnDataSource.Item(“Value2”)
  columns.Item(ctrlUnBnd3.bookmark) =
                                       MyOwnDataSource.Item(“Value3”)

'The following value being set will be ignored, since ctrlBnd will be a bound column
  columns.Item(ctrlBnd.bookmark) = MyOwnDataSource.Item(“Value4”)
End Sub
```