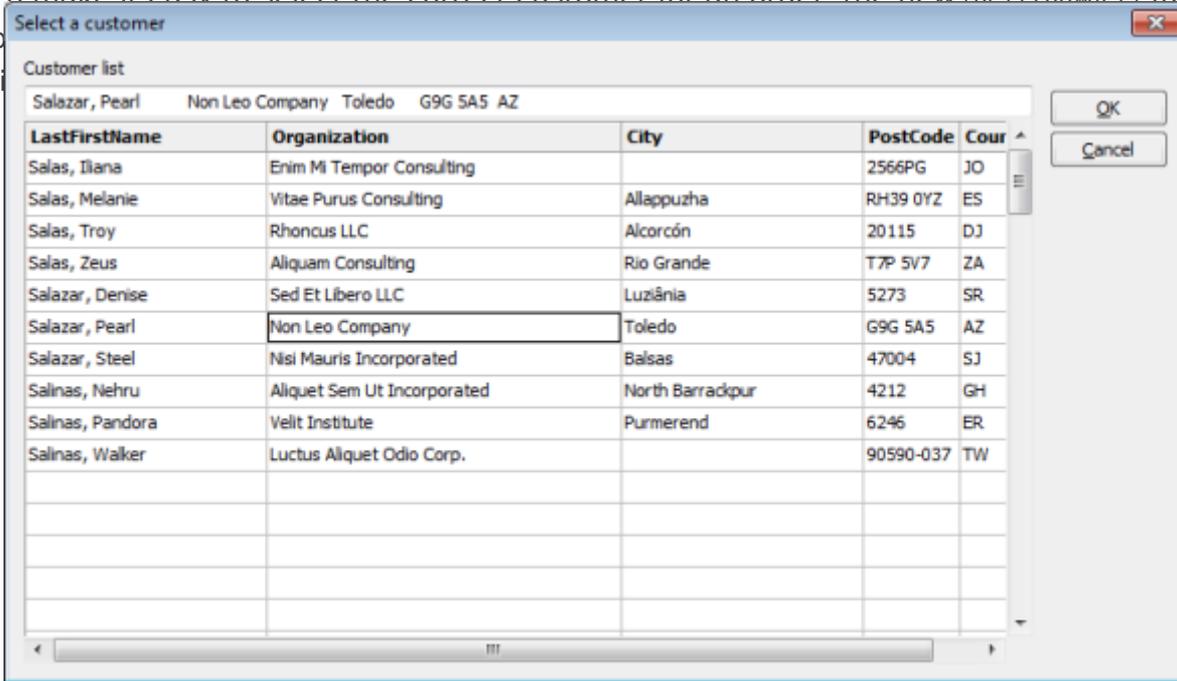


Using the drilldown()

To make it easy to select the correct customer for an order, the new `drilldown()` function was used as a method of



The drill down

window searching for a customer

As the user types in the top edit control, the system detects that and searches against the index passed as the search index. It then fills the grid with data up to the maximum number of desired rows. The code that does this is shown here:

Example 7.4. The findcustomer() Function for the Orders Form

```
function findcustomer(dataform1button me, appwindow appw)
    type(dblrecord) r
    type(dblindex) idx
    sbapplication app
    tdisplayformats dispfmt
    array dispflds, colwidths
    integer e

    if not me.form.preventfocus
```

```

app =@ appw.app
dispfmt =@ app.displayformats
idx =@ app.tables.customer!LastFirstName.index
dispflds =@ array.new()
dispflds[1] =@ app.tables.customer!LastFirstName
dispflds[2] =@ app.tables.customer!Organization
dispflds[3] =@ app.tables.customer!City
dispflds[4] =@ app.tables.customer!PostCode
dispflds[5] =@ app.tables.customer!CountryCode

colwidths =@ array.new()
colwidths[1] = 150
colwidths[2] = 220
colwidths[3] = 150
colwidths[4] = 60
colwidths[5] = 50

e = 0
r =@ drilldown(appw.w, 730, 400, idx, 100, 1, "Select a \
        customer", "Customer list", \
        dispflds=dispflds, colwidths=colwidths, \
        defboolean=dispfmt.defboolean, \
        definteger=dispfmt.definteger, \
        defnumber=dispfmt.defnumber, \
        defdate=dispfmt.defdate, \
        deftime=dispfmt.deftime, \
        defdatetime=dispfmt.defdatetime, \
        datelocale=app.SBLlocale.datelocale, \
        numlocale=app.SBLlocale.numlocale, error=e)

if r != .nul
    updatecustomorderform(me.form, r, appw)
end if
end if
end function

```

The code should be fairly obvious, we first check to make sure we are in data-entry mode by testing the `me.form.preventfocus` value. If it is equal to `.true` then we are not in data-entry, so ignore clicks.

Note

In `preventfocusmode` in the `ongotfocus` event of the `dataform1` controls if the `preventfocus` property is equal to `.true` then nothing happens and focus is ignored. This does not prevent the `onClick` event of buttons from firing, however, so if they should not fire at all times it is necessary to test for the state of this property.

To reduce the amount of typing we declared the `dispfmt` variable and assigned the `app.displayformats` property to it. We then acquire a reference to the index object we wish to use for searching, produce an array to hold the field object references for the fields we wish to display in the grid, and assign the column widths to the `colwidths` array. The `colwidths` array is optional. If they are not passed the column widths will be derived from the table information. Finally the call is made to the `drilldown()` function and if the user clicks on OK, then it will return the selected record object, otherwise it will return `.nul`.

Revision #6

Created 15 November 2021 18:51:09 by Eva Crawford-McKee

Updated 27 December 2021 21:20:45 by Nikolaus Zolnhofer