

# UX - Enhancements

Team Developer 7.4.x

15 03 2022

| Helmut Reimann | Lead Solution Consultant

# Agenda

- Grid
  - Data Export / Import
  - Row Swapping
- Richtext
  - Pasting formatted / unformatted Text
- Tab Control enhancements
- MDI Windows
- Menus

# UX - Enhancements

## Grid – Data Export

---

### SalGridDataExportEx (hwnd, sFileName, nFormat, bVisible, nFlags)

- hwnd – Name or handle of the grid
- sFileName – STRING\_Null shows **Save File Dialog**
- nFormat – System variable „Format\_EXCEL“, „Format\_CSV“ etc.
- bVisible – Visible columns only (**TRUE**)
- nFlags – Combination of **EXPORT\_..** flags

**Note:** Format\_EXCEL exports to \*.xlsx format. If file name is set with „Name.xls“: Compatibility Mode is set: „Old MS-Excel format“

## Code

---

```
◇ Call SalGridDataExportEx( grid1, STRING_Null, Format_EXCEL , FALSE,  
    EXPORT_SelectedRows | EXPORT_SelectedCols |  
    EXPORT_AutoFitColumns | EXPORT_AutoFitRows )
```

# UX - Enhancements

## Grid – Row Swapping

### SalTblSwapRows(hWnd, nRow1, nRow2)

- hWnd – Name or handle of the grid or table
- nRow1 – 1st row to swap
- nRow2 - 2nd row to swap

## Code

```
◆ Pushbutton: pbSwapRows
  ◆ Message Actions
    ◆ On SAM_Click
      ◆ Call SalTblSwapRows( grid1, dfFirstRow, dfSecondRow )
```

# UX - Enhancements

## Grid – Row Count

---

### **nRows = SalTblQueryRowCount(hWnd)**

- hWnd – Name or handle of the grid or table
- nRows – Number of rows in the table or grid

## Code

---

```
◇ Set dfRows = SalTblQueryRowCount( grid1 )
```

# UX - Enhancements

## Richtext - Control

---

### SalRTFSetReadOnly(hWnd, bSet)

- hWnd – Name or handle of the RTF control
- bSet -        **TRUE** = Read only set  
                  **FALSE** = Read only not set

## Code

---

```
◆ Pushbutton: pbTRFSetReadOnly
  ◆ Message Actions
    ◆ On SAM_Click
      ◇ Call SalRTFSetReadOnly( rtc1, TRUE )
```

```
◆ Pushbutton: pbTRFSetReadOnlyF
  ◆ Message Actions
    ◆ On SAM_Click
      ◇ Call SalRTFSetReadOnly( rtc1, FALSE )
```

# UX - Enhancements

## Richtext - Control

---

### bRTF = SaIRTFGetReadOnly( rtc1 )

- bRTF – boolean; Return value TRUE or FALSE
- Rtc1 – Name or Handle of the RTF control

## Code

---

```
◆ Pushbutton: pbRTFGertReadOnly
  ◆ Message Actions
    ◆ On SAM_Click
      ◇ Set bRTF = SaIRTFGetReadOnly( rtc1)
      ◆ If bRTF = TRUE
        ◇ Set dfRTF = 'TRUE'
      ◆ If bRTF = FALSE
        ◇ Set dfRTF = 'FALSE'
```

# UX - Enhancements

## Richtext - Control

---

### On SAM\_RTFPaste

New *Paste* event for RTF controls

“Display a dialog if you want to paste data into a Richtext control”.

You can choose between **data only** or **formatted data**

## Code

---

```
◆ Rich Text Control: rtc1
  ◆ Message Actions
    ◆ On SAM_RTFPaste
      ◆ Set bClipboardHasRTF = wParam
      ◆ If bClipboardHasRTF
        ◆ Set nRet = SalMessageBox("Paste as unformatted text?", "RTF Paste Options",
                                MB_YesNoCancel | MB_IconQuestion)
        ◆ If nRet = IDCANCEL
          ◆ Return FALSE
        ◆ Else If nRet = IDYES
          ◆ Return TRUE
      ◆ Return
```



# UX - Enhancements

## Richtext - Control

- Added language specific ctrl-key accelerators for bold, italic and underline
- Prevent showing default tooltip when **FALSE** is returned from **SAM\_DisplayToolTip**
- Improved touch screen scrolling
- Enable vertical and horizontal scroll when RTC control is set to disable

# UX - Enhancements

## Tab Control

---

### **bEnabled = Sa1TabIsPageEnabled (hTab, nPage)**

- bEnabled – Returns **TRUE** if page is enabled, otherwise **FALSE**
- hTab – Name or handle of the tab control
- nPage – Index of the page (starting at „0“)

## Code

---

```
◇ Set bEnabled = Sa1TabIsPageEnabled( frmResizing.tab1, dfPageNo)
◆ If bEnabled = TRUE
  ◇ Set dflsEnabled = 'Enabled'
◆ If bEnabled = FALSE
  ◇ Set dflsEnabled = 'Disabled'
```

# UX - Enhancements

## MDI Windows

Background Image can now be set / get with:

- `SalPicSetBinary()` | `SalPicGetBinary()`
- `SalPicSetString()` | `SalPicGetString()`
- `SalPicSetImage()` | `SalPicGetImage()`
- `SalPicSetHandle()` can now be used to set the MDI background using a bitmap handle loaded from a DLL

Background image can be cleared by passing:

- `String_Null` to `SalPicSetString()`
- `BINARY_Null` to `SalPicSetBinary()`
- `"0"` to `SalPicSetHandle()`
- `SalPicClear()`

# UX - Enhancements

## MDI Windows

For PNG format images, specify `PIC_FormatPNG` with `SalPicGetBinary( ) / SalPicSetBinary( )` and `SalPicGetString( ) / SalPicSetString( )`, or use `SalPicGetImage( ) / SalPicSetImage( )`

For GIF or JPG format, specify `PIC_FormatObject`, or use `SalPicGetImage( ) / SalPicSetImage( )`

# Demo

**opentext™**

**Q & A**

# opentext™

## Thank you



twitter.com/opentext



linkedin.com/company/opentext

**Get your free TD 7.4 trial now!**

**<https://bit.ly/Gupta-TD-free-trial>**

**[opentext.com/gupta](https://opentext.com/gupta)**