

Modernizing Uniface 9.7 in 10 easy steps

Theo Neeskens

Solution Architect

October 2nd 2015

The logo for Uniface, featuring the word "UNIFACE" in a bold, white, sans-serif font. The letter "A" is stylized, composed of several overlapping triangles in shades of red and white.

Advanced Development Technology

Requirements:


- ▲ New Start Page
 - ▲ White Background
 - ▲ Flat Buttons
 - ▲ Fresh
 - ▲ Different from 9.6
 - ▲ Low cost
-
- ▲ *Blog post and tools to help our customers to modernize their applications:*
<http://unifaceinfo.com/modernizing-uniface-9-7-in-10-easy-steps/>

Start Page




Development ACME Project - [Start]


File Edit View Editors Go To Utilities Help




Mobile



Web



Desktop



Integration

Model
Define the data definitions and default behavior for Uniface components accessing the data

Startup Shell
Define the properties of the startup shell used to execute the application

Form
Define the user interface for Windows applications

Report
Develop components that are used to print data

Service
Separate business from presentation logic or provide functions or calculations that can be used by multiple components


Uniface Community
Links to Uniface resources on the Web

Type	Shortcut	Description	Creation Time
	AAA_0	Shortcut to Form Component Editor: AAA_0 (Form)	25-aug-15 11:50:41

6226 - No permissions found; using default values.

File Edit View Editors Go To Utilities Help

WORKFLOW | INTEGRATION | CONSTRUCTION



DATA ACCESS

BUSINESS LOGIC

PRESENTATION

CONSTRUCTION

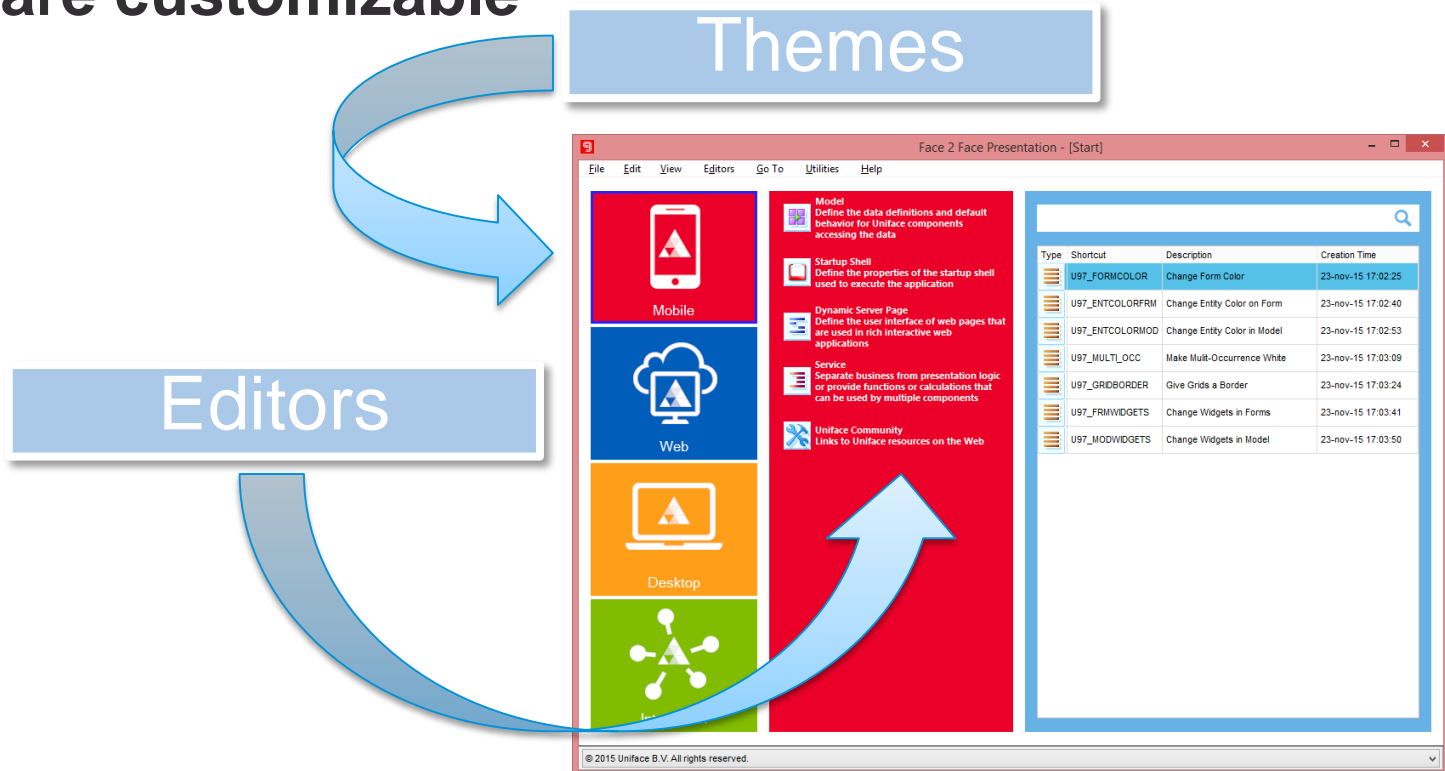
6226 - No permissions found; using default values.

Start Page

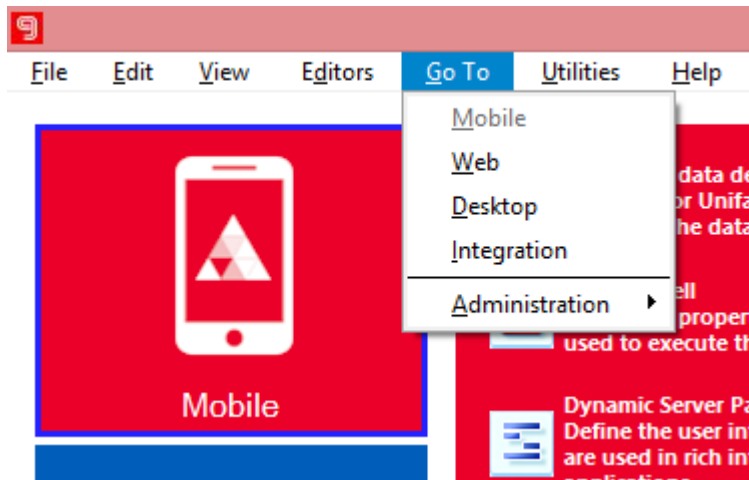
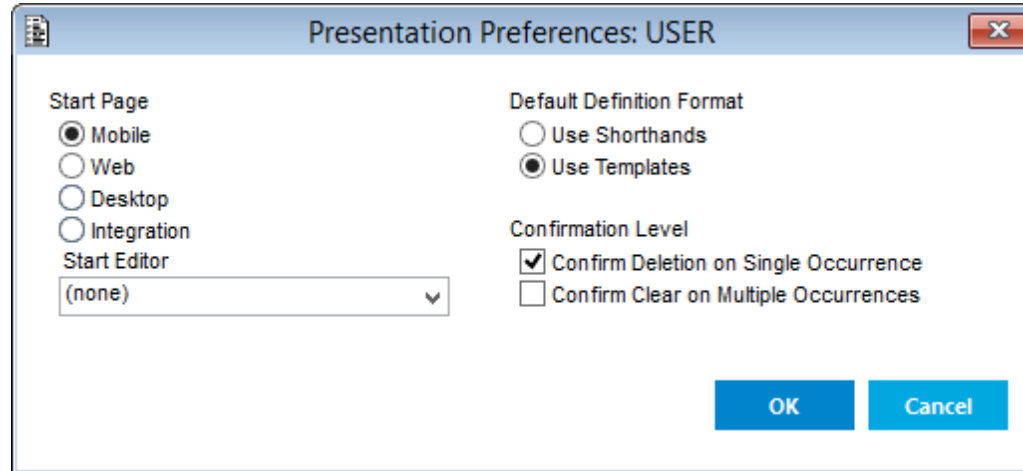
- ▲ Inspiration: Windows 10, 8 styling and color schemes
- ▲ Four Themes: Mobile, Web, Desktop, Integration
- ▲ Editor Shortcuts to most used editors per Theme
- ▲ Re-styled Shortcut area
- ▲ Different!
- ▲ Customizable

Themes

- ▲ **Mobile, Web Desktop and Integration**
Instead of Workflow, Integration and Construction
- ▲ **Per Theme only the most used “Editors”**
- ▲ **Editors are customizable**



Themes



Themes

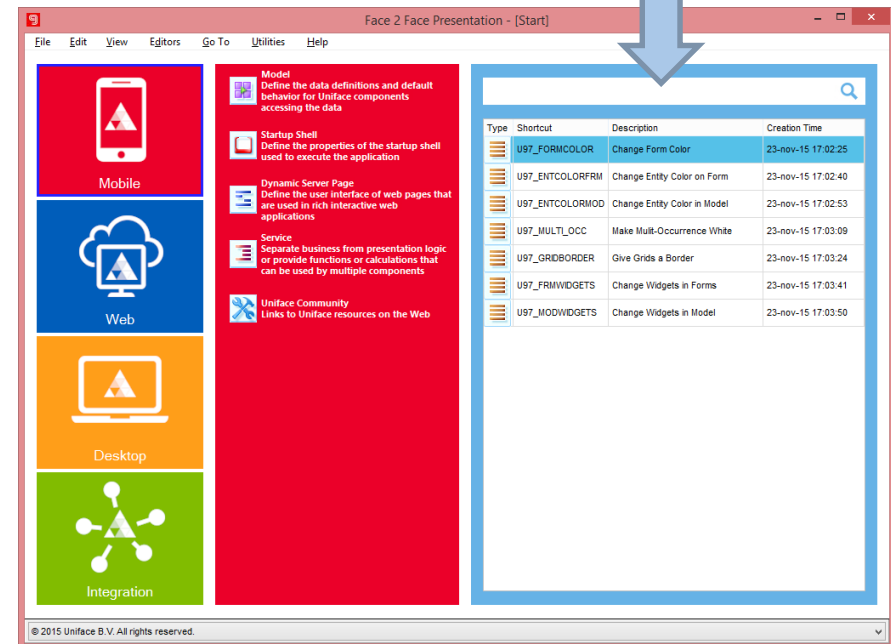
```
startpage.def x
1 <UWWSKSP36ADM>
2 <WKSPCAT.DUMMY>
3 <OCC>
4 <CAT>MOBILE</CAT>
5 <BUTTONTEXT>Mobile</BUTTONTEXT>
6 <BUTTONIMAGE>^U9_MOBILE</BUTTONIMAGE>
7 <TYPES>CONC,APPL,EXTD,EXTS,RESO</TYPES>
8 </OCC>
9 <OCC>
10 <CAT>WEB</CAT>
11 <BUTTONTEXT>Web</BUTTONTEXT>
12 <BUTTONIMAGE>^U9_WEB</BUTTONIMAGE>
13 <TYPES>CONC,APPL,EXTD,EXTW,EXTS,RESO</TYPES>
14 </OCC>
15 <OCC>
16 <CAT>DESKTOP</CAT>
```

```
startpage.def x
25 <TYPES>CONC,APEX,SPEC,EXTW,EXTS,RESO</TYPES>
26 </OCC>
27 </WKSPCAT.DUMMY>
28 <WKSPOBJ.DUMMY>
29 <OCC>
30 <OBJTYPE>APEX</OBJTYPE>
31 <ICON>^U9_SC_APEX</ICON>
32 <NAME>Component Integration</NAME>
33 <DESCRIPTION>Organize components into subsystems</DESCRIPTION>
34 </OCC>
35 <OCC>
36 <OBJTYPE>SPEC</OBJTYPE>
37 <ICON>^U9_SC_SPEC</ICON>
38 <NAME>Signature</NAME>
39 <DESCRIPTION>Define the signatures for Uniface a</DESCRIPTION>
40 </OCC>
41 <OCC>
42 <OBJTYPE>APPL</OBJTYPE>
43 <ICON>^U9_SC_APPL</ICON>
```


Shortcuts

- ▲ Shortcut creation and handling unchanged
- ▲ As many shortcuts as you need!
- ▲ Filtering on Name
- ▲ Sorting on all columns in Grid
- ▲ One set, not one per Theme

Shortcuts



Modernization UDE

Customize – uniface\adm\usys.ini

▲ Theme and Editor Shortcut button fonts

[screen]

IDFCategories=Microsoft Sans Serif,13,regular ;- Theme buttons

IDFButtonText=Segoe UI,Western,8,bold ;- Editor shortcut buttons

▲ Color of shortcut area frame

[areaframes]

SHORTCUTS=uframe(backcolor=#66B2E6;attach=hsize,vsize)

▲ Profile and Shortcut fields properties

[widgets]

IDFSpeedSearch=ueditbox(font=editfont;onedit=T)

IDFTextCell=ueditbox(font=editfont)

Modernization UDE

Customize – uniface\adm\usys.ini

- ▲ Split between common and uniface ini file
- ▲ New IDF logical widgets are not selectable in Field Definition forms.

Not to be used by customer apps, we can change or remove these.

To unhide the secret widgets:
[developer]

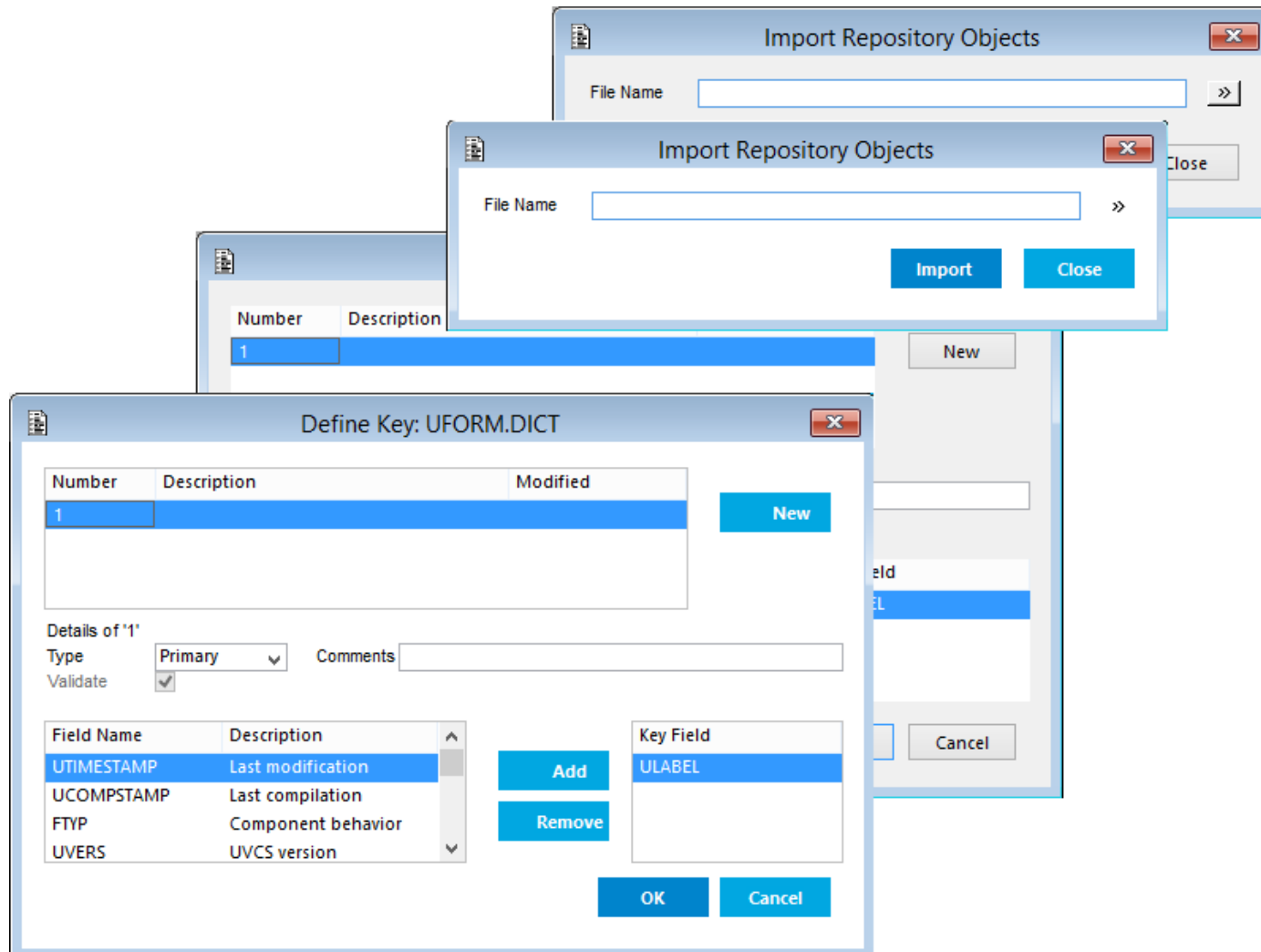
ShowFilteredWidgets = IDF

Colors and Buttons

and menus and panels



White Background



White Background

Seems simple, just change INI setting:

```
[application]
```

```
window=uwindow (backcolor=white)
```

```
shell=ushell (backcolor=#F9FCFF)
```

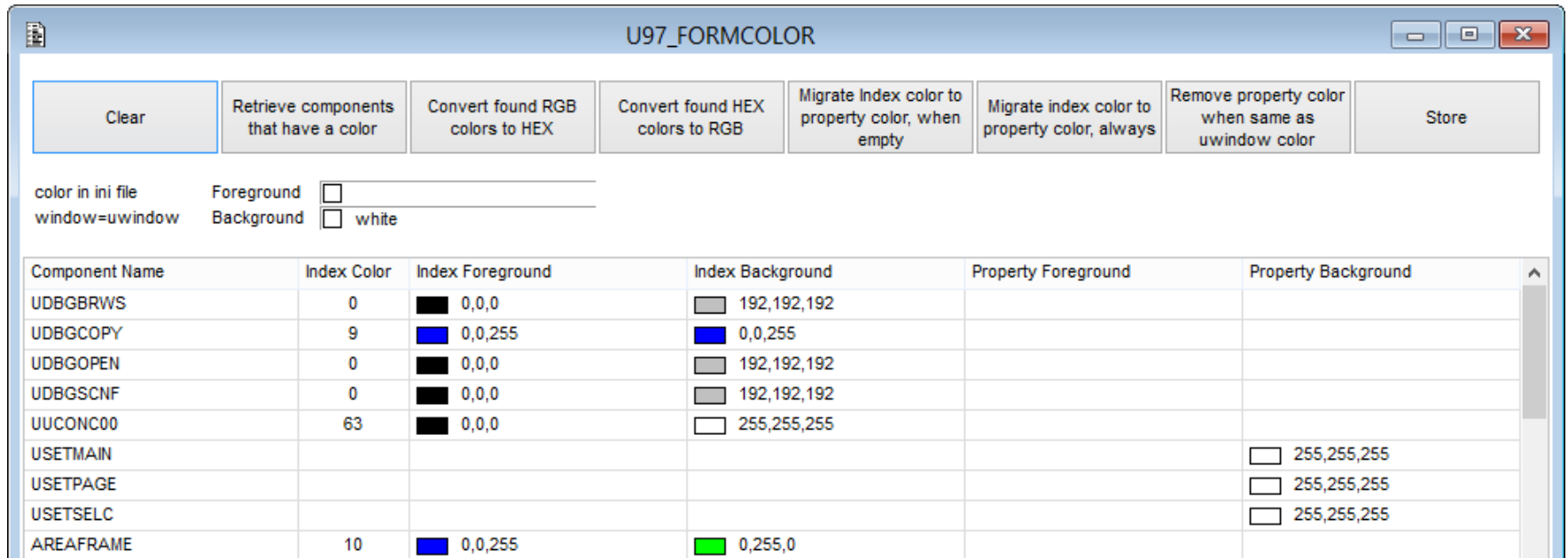
```
menu=umenu (backcolor=white;forecolor=black;backcolorselect=#0084CC;forecolorselect=white;backcolorfill=flat)
```

```
panel=upanel (backcolor=white;backcolorhover=white;backcolorlocked=#55C1E8;backcolorselect=#97D5EC;bordercolorhover=#55C1E8;bordercolorlocked=#0084CC;bordercolorselect=#0084CC)
```

White Background

But:

- ▲ Step 1: Some Forms have a color set
Remove that!



The screenshot shows a window titled "U97_FORMCOLOR" with a toolbar containing buttons: Clear, Retrieve components that have a color, Convert found RGB colors to HEX, Convert found HEX colors to RGB, Migrate Index color to property color, when empty, Migrate index color to property color, always, Remove property color when same as uwindow color, and Store.

Below the toolbar, there are checkboxes for "color in ini file" and "window=uwindow". To the right, there are input fields for "Foreground" and "Background" with a "white" label next to the Background field.

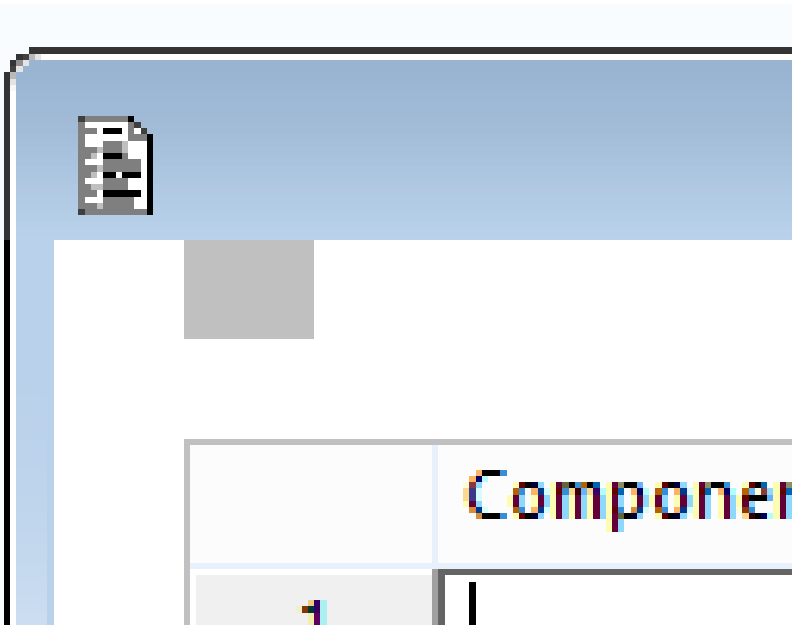
The main part of the window is a table with the following columns: Component Name, Index Color, Index Foreground, Index Background, Property Foreground, and Property Background. The table contains data for various components, including UDBGBRWS, UDBGCOPY, UDBGOPEN, UDBGSCNF, UUCONC00, USETMAIN, USETPAGE, USETSEL, and AREAFRAME.

Component Name	Index Color	Index Foreground	Index Background	Property Foreground	Property Background
UDBGBRWS	0	0,0,0	192,192,192		
UDBGCOPY	9	0,0,255	0,0,255		
UDBGOPEN	0	0,0,0	192,192,192		
UDBGSCNF	0	0,0,0	192,192,192		
UUCONC00	63	0,0,0	255,255,255		
USETMAIN					255,255,255
USETPAGE					255,255,255
USETSEL					255,255,255
AREAFRAME	10	0,0,255	0,255,0		

White Background

But:

- ▲ Step 2: Some Entities on Forms have a color set
Remove that!
- ▲ Step 3: Some Modeled Entities have a color set
Remove that!

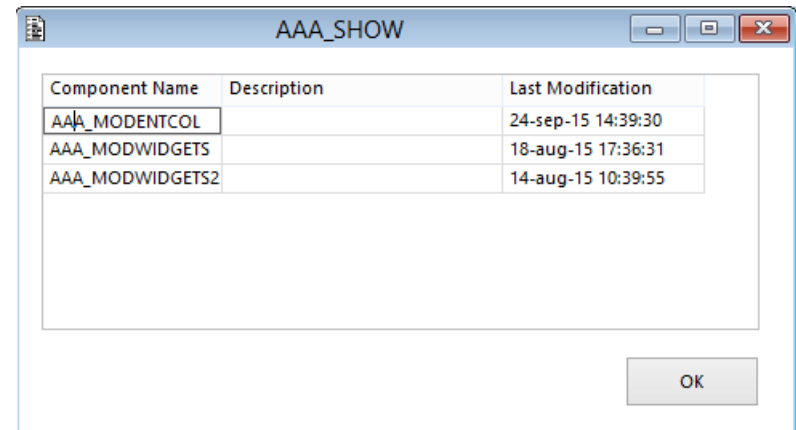
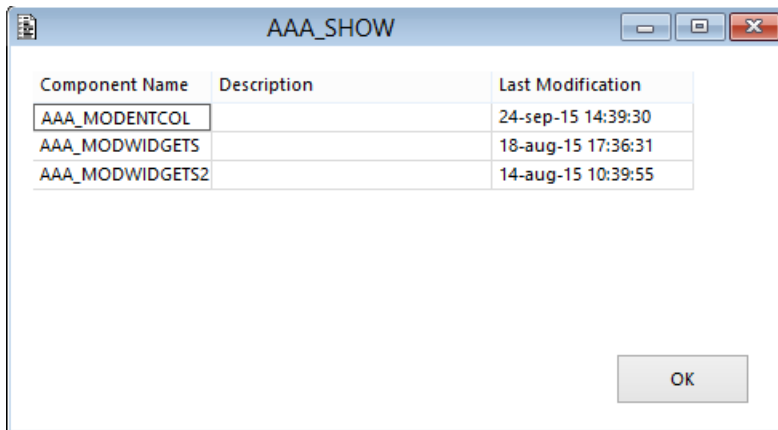


White Background

But:

- ▲ Step 4: Grid widget “disappears” (white on white)
Apply properties to give it a border (new in 9.7)

BorderType=Flat
BorderColor=Silver



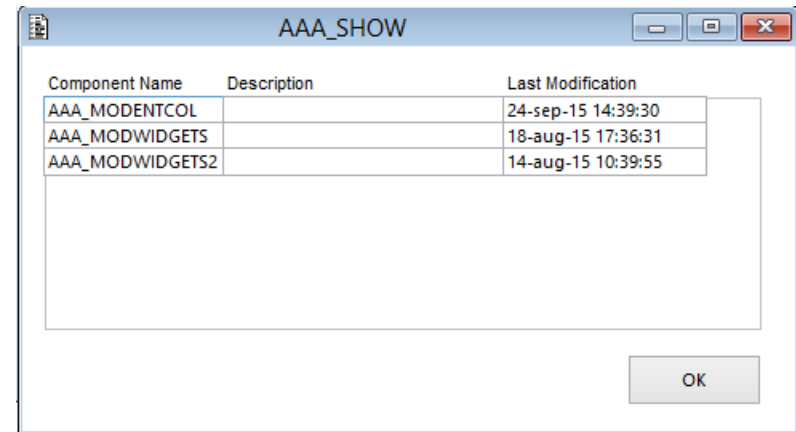
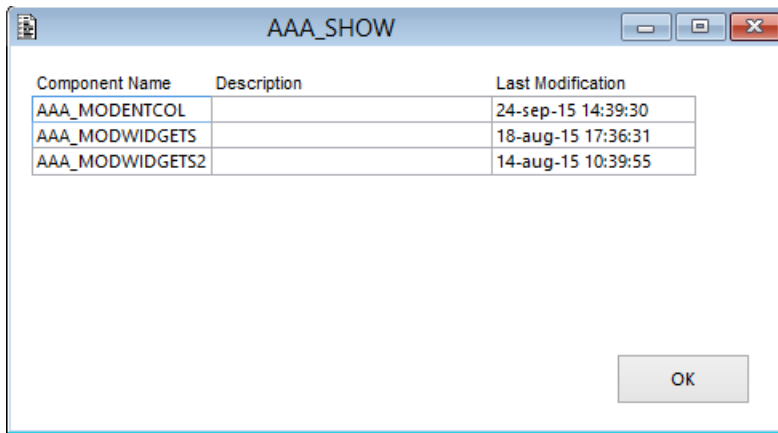
White Background

But:

▲ Step 5: “Simulated grids” made from normal Entities disappear

Make these white again and apply border properties (new in 9.7)

BorderType=Flat
BorderColor=Silver



New entity properties:

BorderColor, BorderType, BorderRadius, DropShadowColor, BackColorStart, BackColorFill, GradientStart, Attach, AttachMargin

Flat Buttons

The diagram illustrates two different button styles in a user interface. The top window, representing a modern design, features flat buttons. It includes a sidebar with three items: 'Interface', 'Syntax', and 'Layout', each accompanied by a small icon. To the right of the sidebar is a table with three rows, each containing a 'Template' dropdown menu and an empty text input field. To the right of each input field is a double-right-arrow button. Below the table is a large 'Comments' text area. At the bottom of this window are two flat buttons: 'OK' and 'Cancel'. The bottom window, representing an older design, features 3D buttons. It has a similar sidebar and table structure, but the double-right-arrow buttons and the 'OK' and 'Cancel' buttons at the bottom have a 3D effect with shadows and highlights.

Interface	Template		>>
Syntax	Template		>>
Layout	Template		>>

Comments

OK Cancel

Flat Buttons

Seems simple, just change INI setting, but:

▲ Step 7: Determine types of buttons Split Command Button in five logical widgets



- ▲ IDFButtonBottom, for the big text buttons at the bottom of Forms
- ▲ IDFButtonSide, for the big text buttons at the right-hand side of Forms
- ▲ IDFButtonSpecial, for the buttons that do not fall in any of the other categories
- » ▲ IDFButtonImage, for the very small buttons with an image on them, like the >> button
- ▲ IDFButtonHeader, for the buttons that form the headers of simulated grids

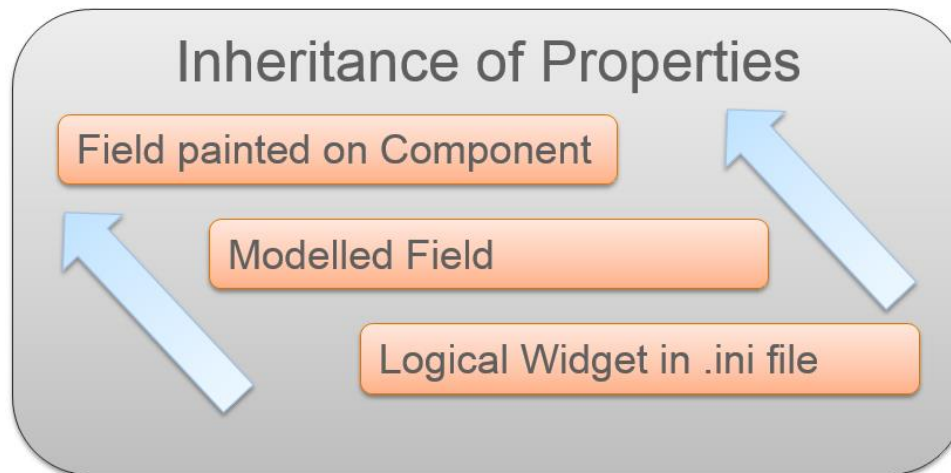
One Entity	>> Many Entity	>> Description
UFORM	UAPLFRM	
UFORM	UXREGS	



Flat Buttons

Seems simple, just change INI setting, but:

- ▲ Step 8: Match modeled widgets with painted widgets
Put most used logical widget in model
- ▲ Step 9: Set properties on painted buttons
Remove as much as possible, to enable control via .ini
- ▲ Step 10: Set properties on modeled buttons
Remove as much as possible, to enable control via .ini



Flat Buttons

Widget Font (default) >>

Label Font (default) >>

More... OK Cancel

Widget Font (default) >>

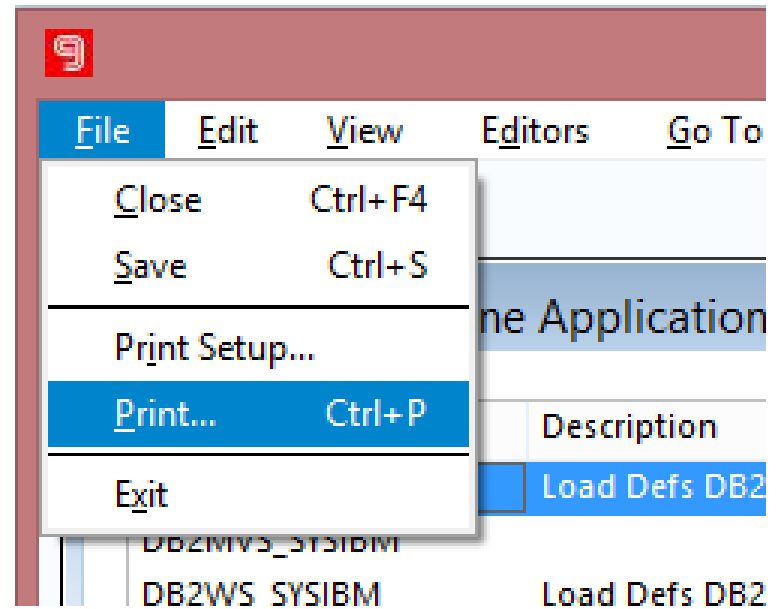
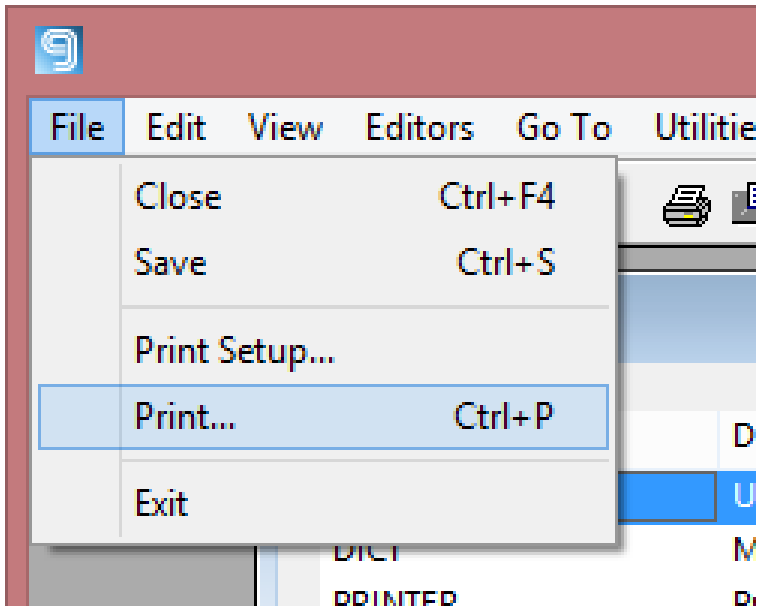
Label Font (default) >>

More... OK Cancel

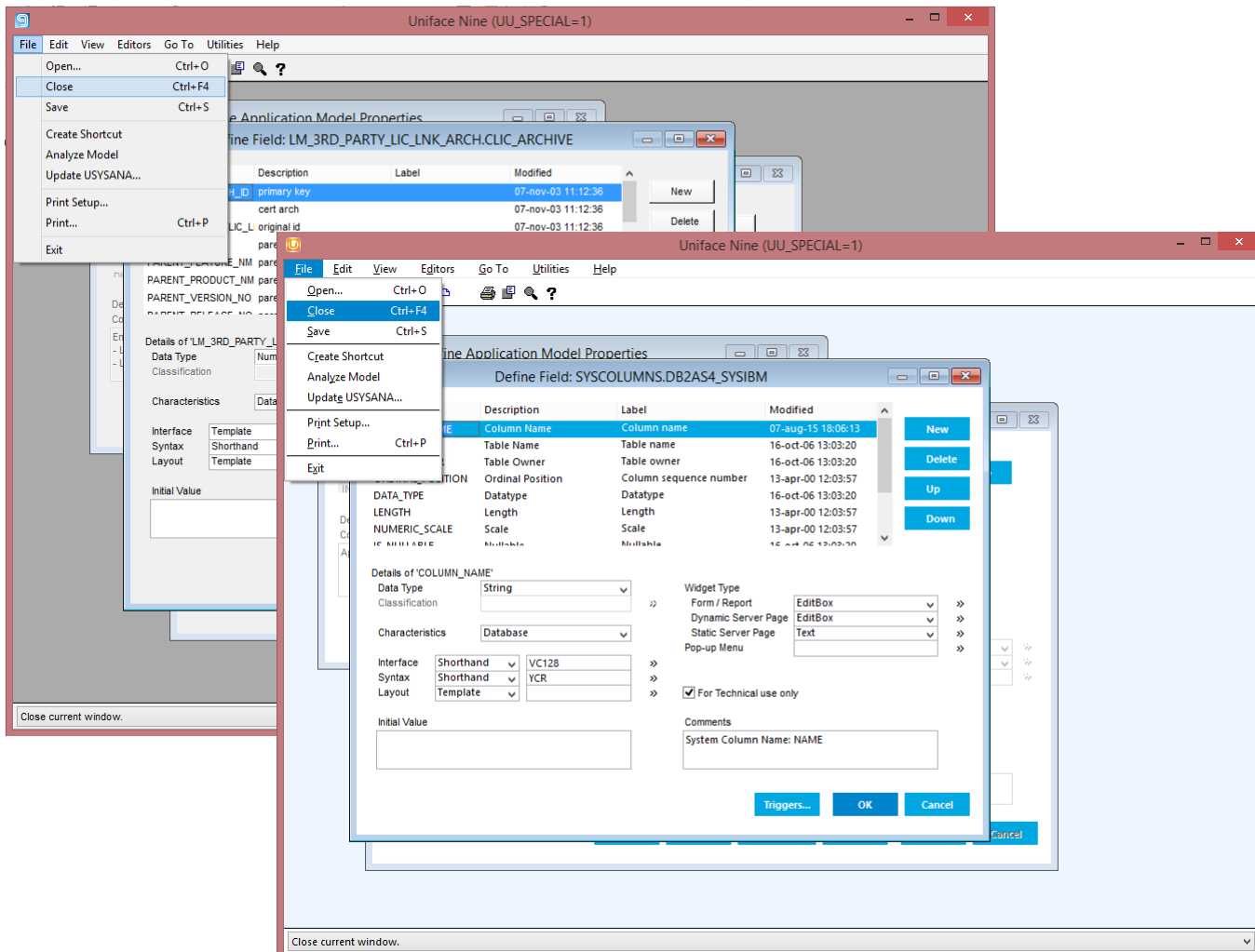
Fresh

 Blue colors were selected for:

- ▲ Buttons
- ▲ Menus
- ▲ Panels
- ▲ Application Shell



Different from 9.6



Details

See [blog post on uniface.info](https://uniface.info) for:

- ▲ More details on each step
- ▲ Download tools for each step

DISCLAIMER

The tools described in this posting are not supported Uniface software. You can download them and use them, modify to your own taste and use them at your own risk. You need the DICT model to be present in your Repository before you can compile and use the tool. Be absolutely sure you have a backup of your dictionary before using any of these tools! You can download the tools [here](#). If you make an improvement to these tools that might be useful to the community, please share it.

One more thing...



Set IDF Title Bar

▲ Set the title of your IDF in your ASN file

[logicals]

IDFTITLE = Development ACME Project

Thank You!

Q&A

More details:

<http://unifaceinfo.com/modernizing-uniface-9-7-in-10-easy-steps/>

UNIFACE

Advanced Development Technology

Follow us online



uniface.com