# Comsoft7

## BoTpl Free Templates

4.1.0

Bo's Template Tamers

By Bo Schmitz

bo@comsoft7.com

# Table of Contents

# Index 0

# 1      Introduction to BoTpl
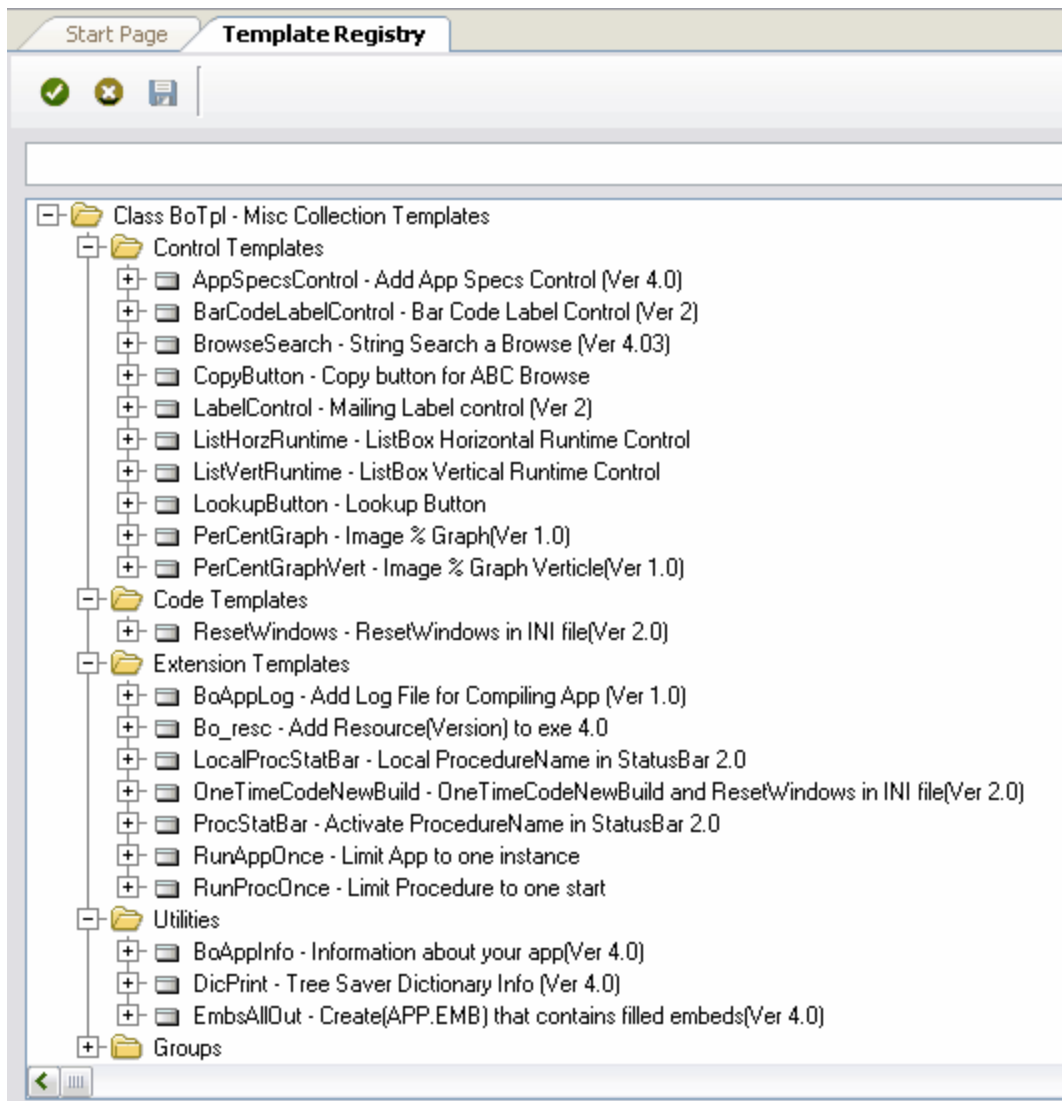
**BoTpl is a Free Template set containing 7 Extension, 10 control, 1 code template, and 3 Utility templates.**

**Some Templates are Legacy and ABC, Some only Work in ABC.  Some work in C4-C7, some only in C55 on.**
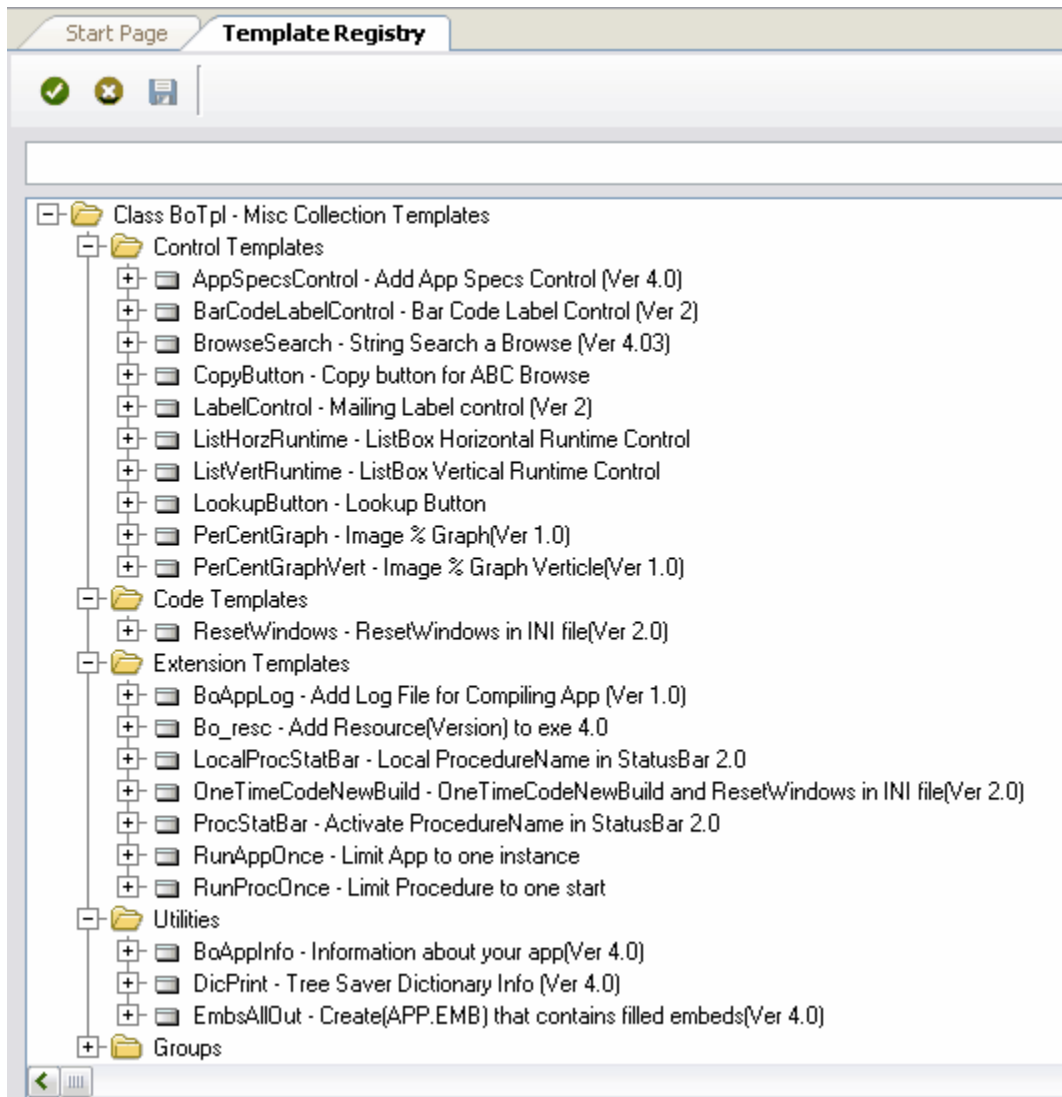


## 1.1     OverView

**BoTpl is a Free Template set containing 7 Extension, 10 control, 1 code, and 3 Utility templates.**

Start Page / **Template Registry**

✓ ✗ 🖫

```
☐─📁 Class BoTpl - Misc Collection Templates
   ☐─📁 Control Templates
      ⊞─☐ AppSpecsControl - Add App Specs Control (Ver 4.0)
      ⊞─☐ BarCodeLabelControl - Bar Code Label Control (Ver 2)
      ⊞─☐ BrowseSearch - String Search a Browse (Ver 4.03)
      ⊞─☐ CopyButton - Copy button for ABC Browse
      ⊞─☐ LabelControl - Mailing Label control (Ver 2)
      ⊞─☐ ListHorzRuntime - ListBox Horizontal Runtime Control
      ⊞─☐ ListVertRuntime - ListBox Vertical Runtime Control
      ⊞─☐ LookupButton - Lookup Button
      ⊞─☐ PerCentGraph - Image % Graph(Ver 1.0)
      ⊞─☐ PerCentGraphVert - Image % Graph Verticle(Ver 1.0)
   ☐─📁 Code Templates
      ⊞─☐ ResetWindows - ResetWindows in INI file(Ver 2.0)
   ☐─📁 Extension Templates
      ⊞─☐ BoAppLog - Add Log File for Compiling App (Ver 1.0)
      ⊞─☐ Bo_resc - Add Resource(Version) to exe 4.0
      ⊞─☐ LocalProcStatBar - Local ProcedureName in StatusBar 2.0
      ⊞─☐ OneTimeCodeNewBuild - OneTimeCodeNewBuild and ResetWindows in INI file(Ver 2.0)
      ⊞─☐ ProcStatBar - Activate ProcedureName in StatusBar 2.0
      ⊞─☐ RunAppOnce - Limit App to one instance
      ⊞─☐ RunProcOnce - Limit Procedure to one start
   ☐─📁 Utilities
      ⊞─☐ BoAppInfo - Information about your app(Ver 4.0)
      ⊞─☐ DicPrint - Tree Saver Dictionary Info (Ver 4.0)
      ⊞─☐ EmbsAllOut - Create(APP.EMB) that contains filled embeds(Ver 4.0)
   ⊞─📁 Groups
```

**AppSpecsControl** is a control to drop into a window to list crucial information on the build, such as the app name, dictionary used, and their versions. It has a version and build number that is synchronized with Bo_resc template or can stand alone with its own auto increment build feature.

**BarCodeLabelControl** is a control to drop on a report detail that configures dynamic sized labels with barcode for things such as labeling inventory items. You can control label size and fonts from the included Labels file.

**BrowseSearch** is a control template to drop on a window with an ABC Browsebox to allow searching records and memos on a file in an ABC Browsebox. It does not affect current ranges or filters, but does additional filtering in takerecord embed.

**CopyButton** is a control button designed to copy a record from an ABC Browsebox and

children to a cloned record.

**LabelControl** is a control to drop on a report detail that configures dynamic sized address labels.  Now includes a salutation line and more expression prompts to build your own input. You can control label size and fonts from the included Labels file.

**ListHorzRuntime** is a drop in control to dynamically change the runtime properties of a listbox. It changes the font and background color of the listbox you select. It contains a color selector button, font selector button, and save, apply, and default buttons. It is configured in a horizontal layout.

**ListVertRuntime** is a drop in control to dynamically change the runtime properties of a listbox. It changes the font and background color of the listbox you select. It contains a color selector button, font selector button, and save, apply, and default buttons. It is configured in a vertical layout.

**LookupButton** is a drop in control to call a select procedure, and assign fields after looking it up.

**PerCentGraph** is a drop in control that uses an image control to create a horizontal bar type graph.

**PerCentGraphVert** is a drop in control that uses an image control to create a vertical bar type graph.

**BoAppLog** is a Global Extension Template that adds the ability to create a log file with your application logging dictionary changed, and procedures changed with each compile. Gives you the ability to add one-time comments to the log file as well.

**Bo_resc** 32bit only. is a global Extension template that adds Version Information Resources to the exe or dll. It includes an auto increment build feature for each compile. The Build Number is stored in %ApplicationBld.txt. It interacts with the **AppSpecsControl** . The defaults are set up for my use, and you can change them in the template editor by editing the DEFAULT Settings, Example --- #PROMPT ('Copyright', @s255), %ResLegalCopyright, DEFAULT('Copyright<169> Comsoft7<174>1985-2009')

**OneTimeCodeNewBuild** is an extension template for the frame that runs once on a new install or on an updated version of the program. Uses Ini Manager (ABC only) and has an embed point for code to run once. Has a checkbox if you want to reset all the window sizes and positions stored in the Ini file. (If you redesigned the screens.) It is activated by manually incrementing or changing a version value. The version value is stored in the App Ini file under ('RWSZ','RWSZFlag').  Can use the code template **ResetWindows** to Reset all windows to original size and position if there are any problems or windows get lost from moving off screen.  Put the code template under a reset button on App Frame toolbar or something similar.

**ProcStatBar** is a Global Extension, and **LocalProcStatBar** is the local extensions that work together to put the name of the active procedure and template type into the status bar of the main frame.

**BoAppInfo** is a utility to create a text file and/or print information about your app. Now includes module data and does an instring search of the txa for all supported image files, lib references, and wav's in your app. Lists all the usual stuff and includes formulas.

**DicPrint** is another dictionary to text/print utility. This one is very concise, and you can configure it to include keys and relationships if you want them. It has the ability to select all files or just the ones you want.

**EmbsAllOut** captures global, module, and procedure embedded code and outputs them to a text file in a format suitable to copying to templates or just cutting and pasting the code in other embeds.

## 1.2 Whats New

**August 16, 2009**
Added BoAppLog template.  Add Log File for Compiling App (Ver 1.0)

**5-12-2009**
LookupButton template. Comment out for Clarion, throws errors. #!ADD(%ProcsCalled,% BsUpdProc)

**Version 4.02**

**4-8-2009**
changed code for BrowseSearch line 72 #OF('LOC:SearchString') ! #OF('SearchString')

**Version 4.01**

**1-10-2009**
Bug Fix changed code for BrowseSearch memos line 162

**Updates for Version 4.0**

**11-5-2008**
Added CopyButton control for ABC browses.

**11-2-2008**
Added RunAppOnce Application Extension using Mutex.
Added RunProcOnce, 'Limit Procedure to one start'

**10-3-2008**
Added multi ability to BrowseSearch Control.

**9-28-2008**
Cosmetic changes to AppSpecsControl**.**
Changed to use %ControlOriginal for list runtime controls.  Allows renaming of controls.

**9-20-2008**
Added SELECT(%BSList) to return focus to list in BrowseSearch
ADDED %(%StripPling(%BSIniFile)) to allow variables for ini file.

**5-3-2007**
Fixed Template Name in Global Headers.
BoAppInfo, Information about your app Ver 2.3  Added Prototype,Parameters, and
Returntype to procedures.

**1-18-2006**
Tree Saver Dictionary Info Ver 2.1 Added file,field,key,Relation count to dictionary out.

**12/9/2005**
EmbsAllOut Version 3.1 Added View output and Mark Orphan embeds

**2/19/2004**
OneTimeCodeNewBuild and ResetWindows in INI file(Ver 2.0)  Changed embed points for
C6.2 for resizing,and to accommodate Pause Control.
Added code template for resize windows to RWSZ.tpw  Forces reset if windows get lost.


**Updates and additions Oct 19, 2003 BoTpl Version**

PerCentGraph is a new control template that uses an image control to create a horizontal
bar
type graph. 10/1/2003 added do bar Image graph based on variable Added per request of
Neville
Wright

PerCentGraphVert is a new control template that uses an image control to create a vertical
bar
type graph.

EmbsAllOut has been updated to allow better control of output. Now you can control wheter
global
and module data get outputted, and restrict/select procedures included in output.

**Updates and additions July 31, 2003**

Bo_resc has been updated to include custom resources.(Multi-line)

BrowseSearch is a new control template to allow searching records and memos on a file in an
ABC Browsebox.

**Updates and additions July 10, 2003**

C6 compatible of course<g>.

Bo_resc has been updated to now work correctly on W2K and XP Operating Systems. Up to
version 99.99.99.99<g>

AppSpecsControl has been modified to get rid of errors when no dictionary is used. Dictionary
appears to be a reserved word now in C6.

BoAppInfo has been modified for more selective choices, and instring search of
%ProcedureCategory and %ProcedureTemplate

New control Templates added

BarCodeLabelControl dynamic resizing and fonts for inventory type application.

LabelControl dynamic resizing and fonts for standard mailing label type applications.

ListHorzRuntime Runtime control over font and background color of listboxes.

ListVertRuntime Runtime control over font and background color of listboxes.

## 1.3    Getting Started

BoTpl is a Free Template set containing 7 Extension, 10 control, 1 code, and 3 Utility templates.

Some Templates are Legacy and ABC, Some only Work in ABC.  Some work in C4-C7, some only in C55 on.

Templates only appear if they are available to use at the time.

Global extension templates appear in the global add extensions window.
Procedure extension templates appear in the procedure level when adding extensions.

Control templates appear on the list when editing windows and adding control templates.

The label and barcode label controls are only available on reports.

The utility templates are available from the ide menu under  Application\Template Utility...
or with Ctrl+U

## 1.4     License

License & Copyright Notice

BoTpl(tm) Template set is copyright 2009 by Comsoft7.
It includes derivations from other authors, and credit
is given when credit is due.

BoTpl(tm) is provided without warrantee or guarantee, implied
or otherwise. Use at your own risk. Using BoTpl(tm) implies
acceptance of this agreement, along with recognition of copyright.
Comsoft7 cannot be held liable for any damages or business losses,
which may occur as a direct or indirect result of using BoTpl(tm).

#! Parts of BoTpl(tm) Template are the intellectual property
#! of Bo Schmitz - AKA Comsoft7 Copyright 1985-2009
#!
#! The author has NOT developed all template code independently,
#!
#! The template is provided for free usage of all developers
#!
#! BoTpl(tm) is all code, please observe the limited copyright.
#! You may modify BoTpl(tm) to suit your purposes.
#! Please advise the Author of any major changes, your needs,
#! and perhaps these can be included as features.
#!
#! BoTpl(tm) can be copied, modified, in part or in whole
#! and distributed to another person as long as it is in
#! Its complete form with credits as is found in the BoTplSU.exe.
#!
#! The author may be contacted by email at bo@comsoft7.com
#!
#! No warranty of any kind! Use at your own risk.
#!

# 2 Control Templates
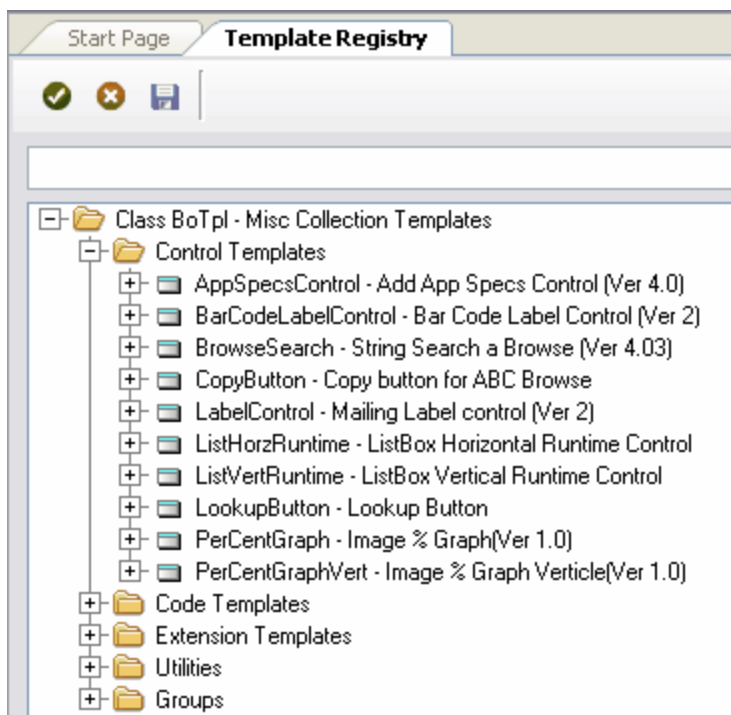


Control templates appear on the list when editing windows and adding control templates. The label and barcode label controls are only available on reports.

## 2.1 AppSpecsControl

AppSpecsControl is a control to drop onto a generic window to list crucial information on the build, such as the app name, dictionary used, and their versions. It has a version and build number that is synchronized with **Bo_resc** template or can stand alone with its own auto increment build feature.

It comes with it's own close button as part of the group structure. Salt to taste.
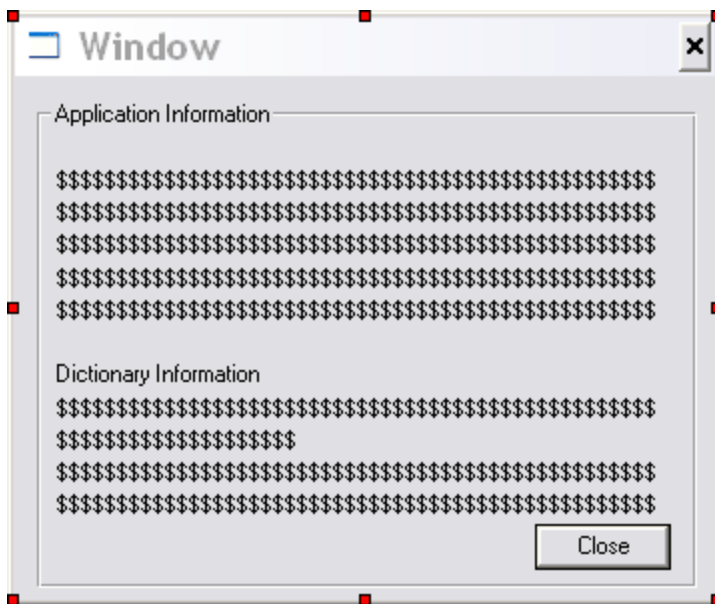
Below are results of sample app.
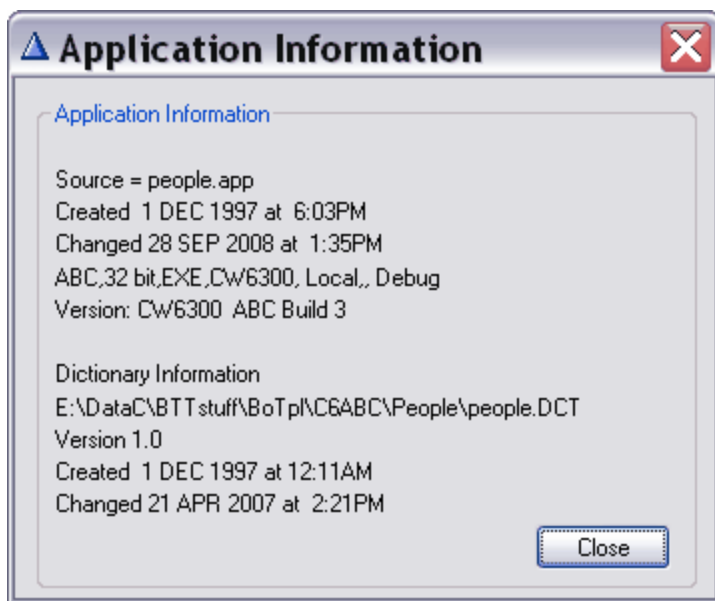


This information is gathered automagically for you.

You can use it stand alone, and put your own build number in it, or use the Enable Build Autoinc feature.  It will also Synch with the Bo_resc template.

Stand Alone

Synch with Bo_resc global template

## 2.2    BarCodeLabelControl

BarCodeLabelControl is a control to drop on a report detail that configures dynamic sized labels with barcode for things such as labeling inventory items. You can control label size and fonts from the included Labels file. This label control uses a barcode font, and a free 3of9.ttf  is included for testing.

Create a report with just a detail band, and drop in the barcode Label control.  If you are using the included Labels file, add it as other files first, and it will auto fill the template

prompts for the label sizes and fonts. A C55 txd file is included with the file definition, as well as the tps file. A C55 TXA (abc) file is also included with the Label browse and update form that you can import into your procedure. This same label file will work for both the barcode labels, and mailing labels included in this template set. See your directories under 3rdParty/ Examples/Comsoft7/BoTpl for the included files and *.tx?'s

Size the detail accordingly, and it will be dynamically resized as needed from the Labels file.
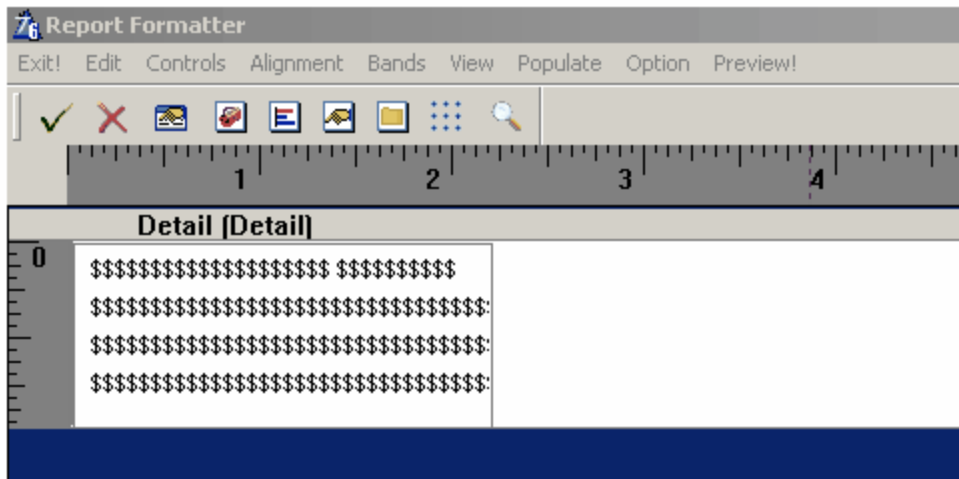


Set your filters, ranges, etc to limit the printing of the labels.
Then fill in the Template prompts.
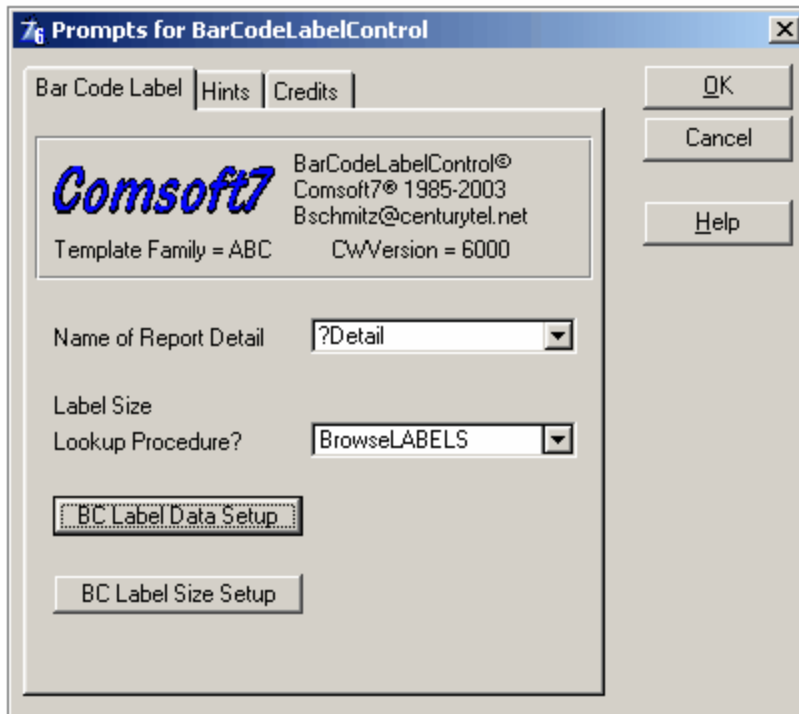
Select the name of the Detail band from the drop down list, and the name of your Label size lookup procedure from the procedure dropdown list. This will autodefault to BrowseLabels if you have included the procedures in the txa file.  Then click on the button For the BC Label Data Setup.
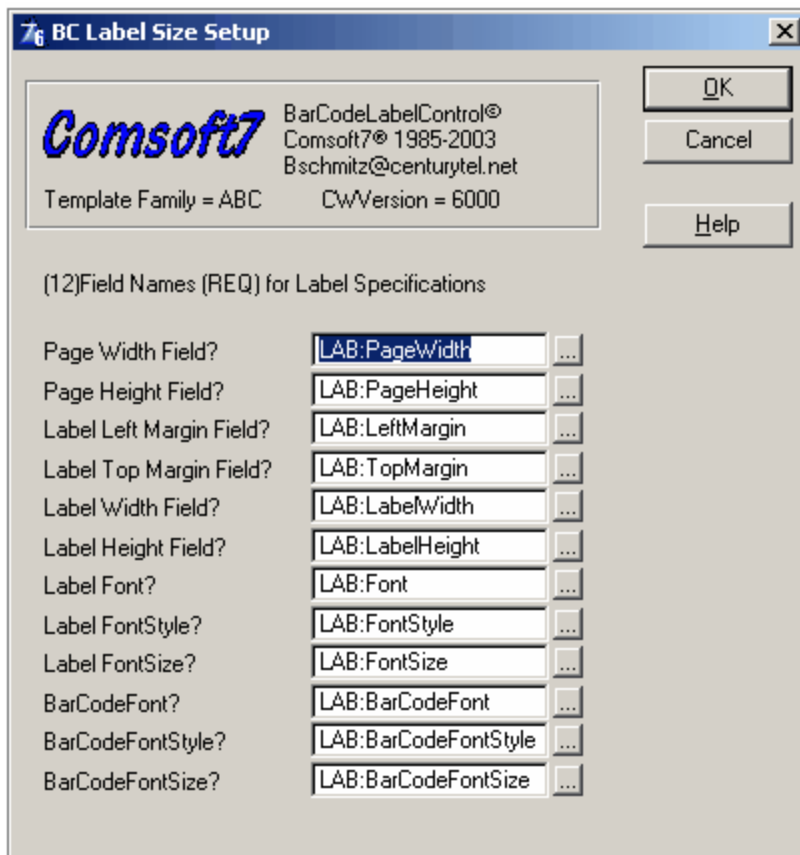


The required part number is put in the top left hand field on the label in man readable's, and is also used for the barcode font field.

Check the Date box if you want a date included on the label run, and select a format for the date. It will appear on the label to the right of the Part Number on the top line.

The next 2 lines can be used for description and extended description of the part, or whatever additional info you want to include.

The last checkbox and prompt is for adding a suffix and prefix to the Barcode font, in this case a '*' is used so barcode readers correctly read the barcode font.

OK this button, and then go to the BC Label Sizes Button of Prompts.

They will be auto filled if you have used the Labels file included with the template.

Otherwise fill in the prompts with the data fields from your label size file.

Observe the results.

## 2.3    BrowseSearch

BrowseSearch is a control template to drop on a window with an ABC Browsebox to allow searching records and memos on a file in an ABC Browsebox. It does not affect current ranges or filters, but does additional filtering in takerecord embed.

It is not advised for filtering a large number of records.

It places strings over the Record and Memos for the file.
It then does an instring search function on those strings for each record.
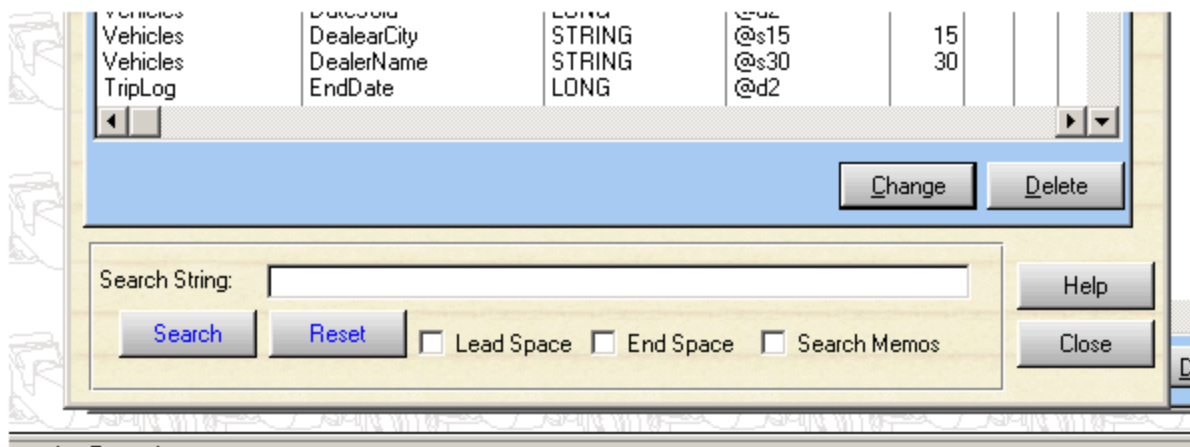


Type the text to search for into the entry control, and further limit the search by adding a lead or end space. Buttons do as they say. Search button limits the browse with new filter in addition to any existing filters and ranges, and reset restores browse to original settings. A switch is provided to optionally search the memo fields.  Include the memo's in browse or make them hot fields.
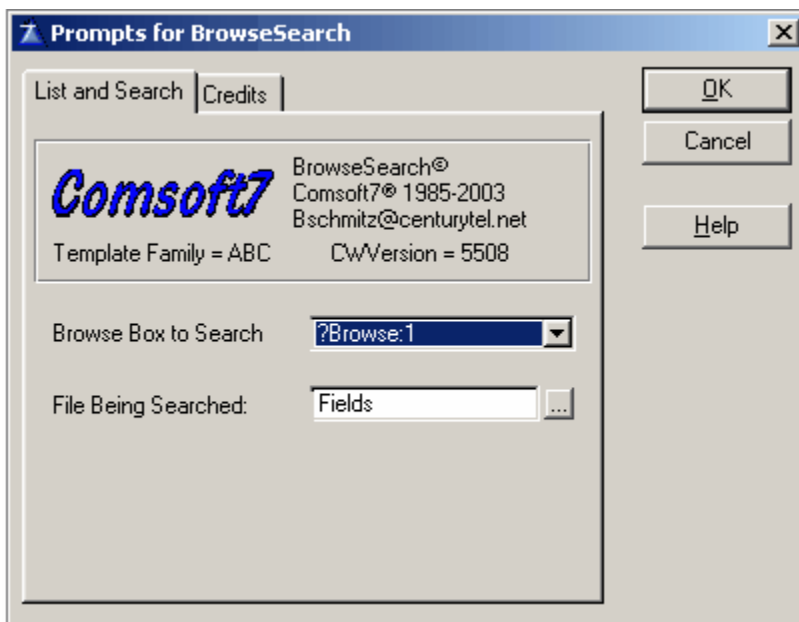


Prompts are straight forward and simple. Select the listbox and file involved.
This is usually the Primary file for the Browsebox<vbg>.

## 2.4    CopyButton

**CopyButton** is a control button designed to copy a record from an ABC Browsebox and children to a cloned record.
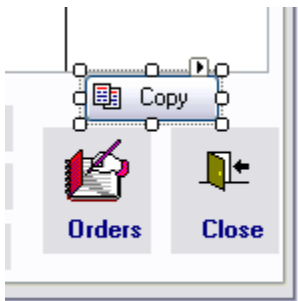


Prompts are pretty straight forward.

All have dropdown selection or lookup buttons.
Select the proper Browse, File, System Id Field, and the insert button for the browse.

Press the insert button under Related Files to copy if you have children to copy.
The files must be in the procedure and have a relationship established.

Selector buttons have been provided for getting the correct info.

## 2.5     LabelControl

LabelControl is a control to drop on a report detail that configures dynamic sized address labels. Now includes a salutation line and more expression prompts to build your own input. You can control label size and fonts from the included Labels file.

Create a report with just a detail band, and drop in the Label control.
If you are using the included Labels file, add it as other files first, and it will autofill the template prompts for the label sizes and fonts. A C55 txd file is included with the file definition, as well as the tps file. A C55 TXA (abc) file is also included with the Label browse and update form that you can import into your procedure. This same label file will work for both the barcode labels, and mailing labels included in this template set. See your directories under 3rdParty/Examples/BoTpl for the included files and *.tx?'s
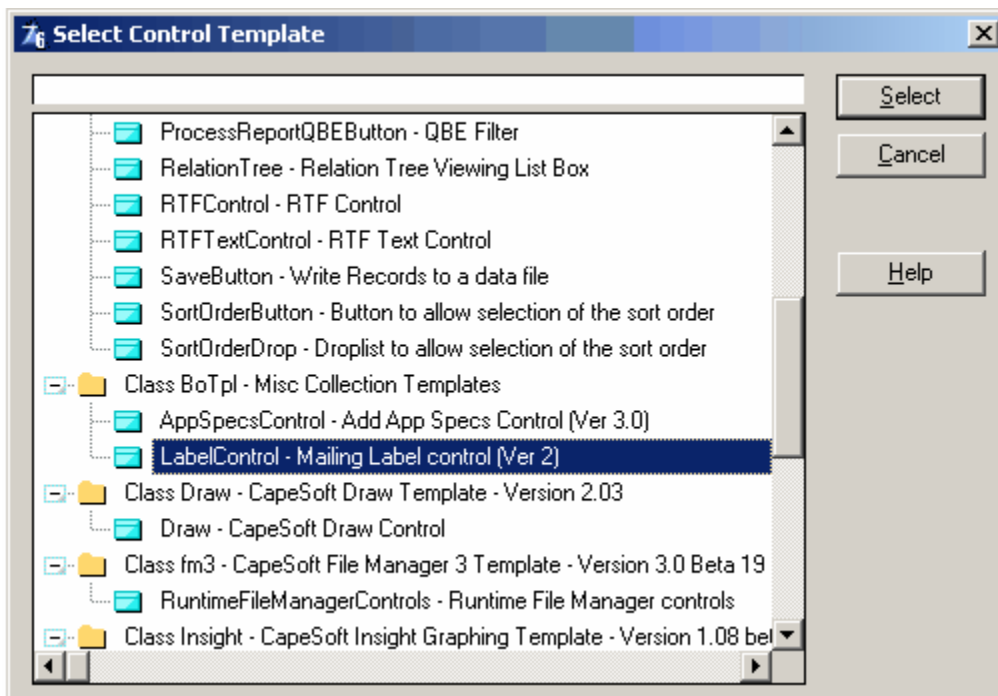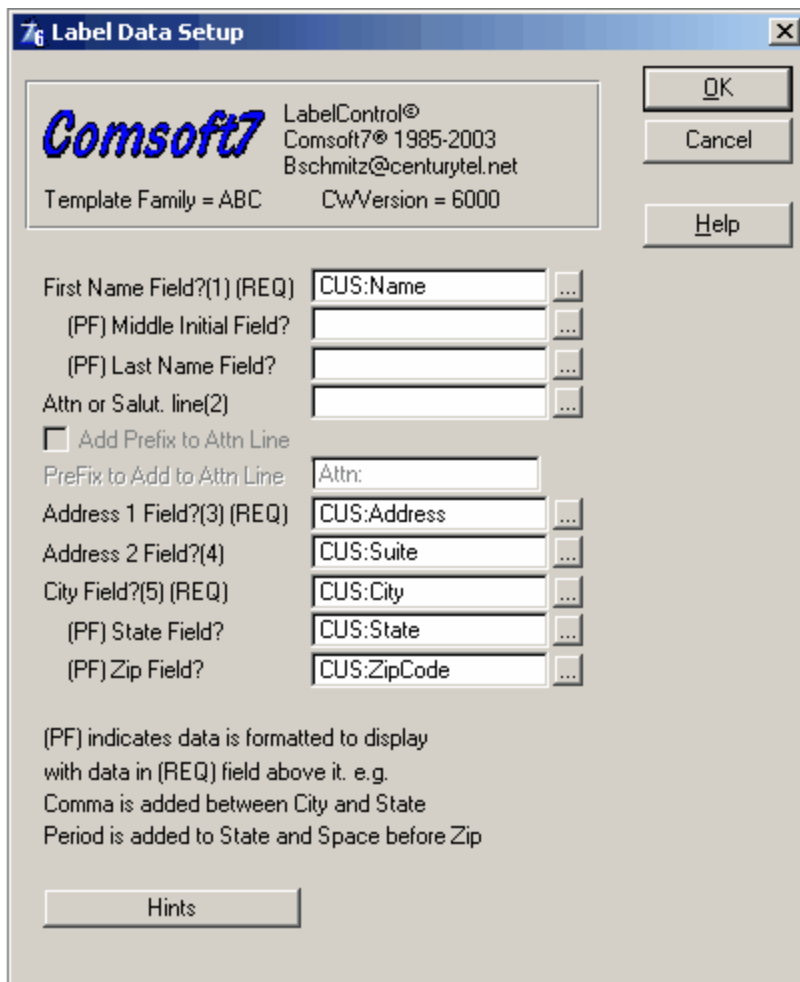


Size the detail accordingly, and it will be dynamically resized as needed from the Labels file.
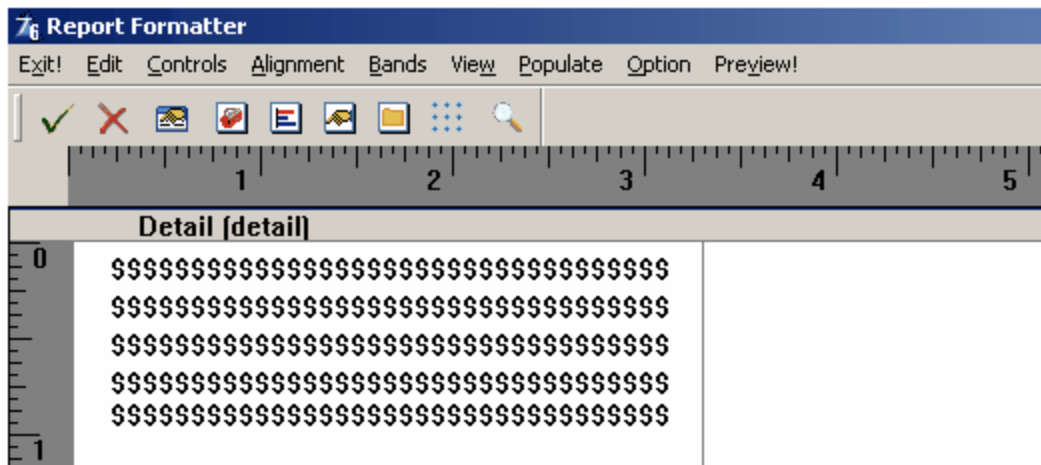
Set your filters, ranges, etc to limit the printing of the labels.
Then fill in the Template prompts.
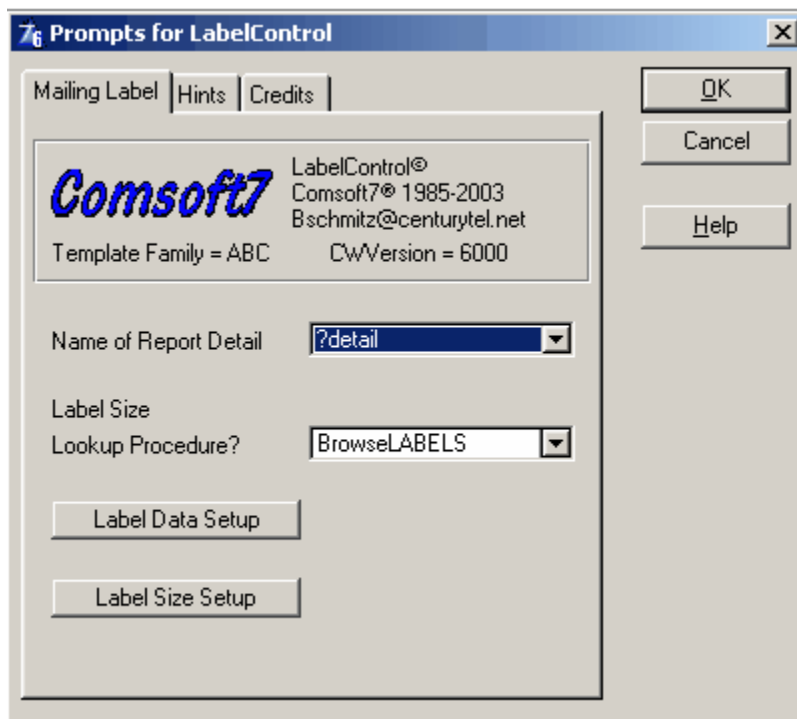


Select the name of the Detail band from the drop down list, and the name of your Label size
lookup procedure from the procedure dropdown list. This will autodefault to BrowseLabels if
you have included the procedures in the txa file.
Then click on the button For the Label Data Setup.

Select the data to be included in the Label Fields.

Note: Three fields are required, the First Name field, and this can include the entire name, because the following 2 fields are preformatted add added to the first line to create a name field if you are setup that way. Address 1 Field is required, and City Field is required. The PF fields format the additional fields if you have them stored separate in the file structure. You can, however use the label anyway you want, and build your own expressions, as the prompts are EXPR fields.

OK this button, and then go to the Label Size Setup Button of Prompts.

They will be auto filled if you have used the Labels file included with the template.

Otherwise fill in the prompts with the data fields from your label size file.

Observe the results.

## 2.6　ListHorzRuntime

ListHorzRuntime is a drop in control to dynamically change the runtime properties of a listbox. It changes the font and background color of the listbox you select. It contains a color selector button, font selector button, and save, apply, and default buttons. It is configured in a horizontal layout.

There is a group handle to move around the controls easily.

It is a multi control, so you can add as many controls as you have listboxes.

you can put the control on a configuration tab behind other information, etc.

| Label Type | Page Width Page Height | Label Width Label Height | Top Margin Left Margin | Font Font Style Font Size | Bar Code Font Bar Code Font Style Bar Code Font Size |
|---|---|---|---|---|---|
| 1 x 2-5/8  Cur | 8.50 10.50 | 2.62 1.00 | 0.62 0.50 | Arial 700 12 | 3 of 9 Barcode 700 24 |
| Avery 2-Up | 8.50 11.00 | 4.00 1.50 | 0.50 0.50 | Arial 400 10 | 3 of 9 Barcode 700 24 |
| Avery 3-Up5351 | 8.50 10.50 | 2.63 1.00 | 0.62 0.50 | Arial 400 8 | 3 of 9 Barcode 700 20 |
| Avery 4-Up | 8.50 11.00 | 2.00 1.00 | 0.25 0.25 | Arial 400 8 | 3 of 9 Barcode 700 18 |
| Avery 8163 2x5 | 8.50 11.00 | 4.00 2.00 | 0.75 0.50 | Arial 700 16 | 3 of 9 Barcode 700 24 |
| Avery 8164 2x3 | 8.50 11.00 | 4.00 3.33 | 1.50 0.50 | Arial 400 20 | 3 of 9 Barcode 700 24 |

Values are stored in an inifile and applied when starting the window.
Buttons call up the standard Windows **Color** Dialogue and **Font** Dialogue.

**Default** will load the orginal design fonts and colors and save them to the ini.
**Apply** will apply the current selections to the listbox, for the current session, but not save them.
**Save Settings** will save the current settings to the ini for this and further uses of the listbox.
You can delete the color control or font control if you only want to give the user partial functions.
You can, of course move the controls individually as needed to position.
The only prompts you need to fill are which listbox to control, and the ini file to use. It defaults to BST.INI so as not to affect the main ini for the app.



## 2.7 ListVertRuntime

ListVertRuntime is a drop in control to dynamically change the runtime properties of a listbox. It changes the font and background color of the listbox you select.
It contains a color selector button, font selector button, and save, apply, and default buttons.
It is configured in a vertical layout.
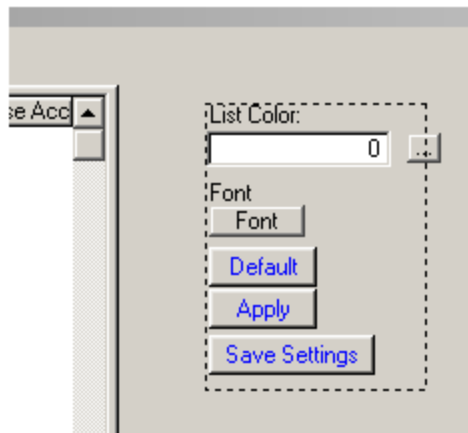
There is a group handle to move around the controls easily.

It is a multi control, so you can add as many controls as you have listboxes.

you can put the control on a configuration tab behind other information, etc.
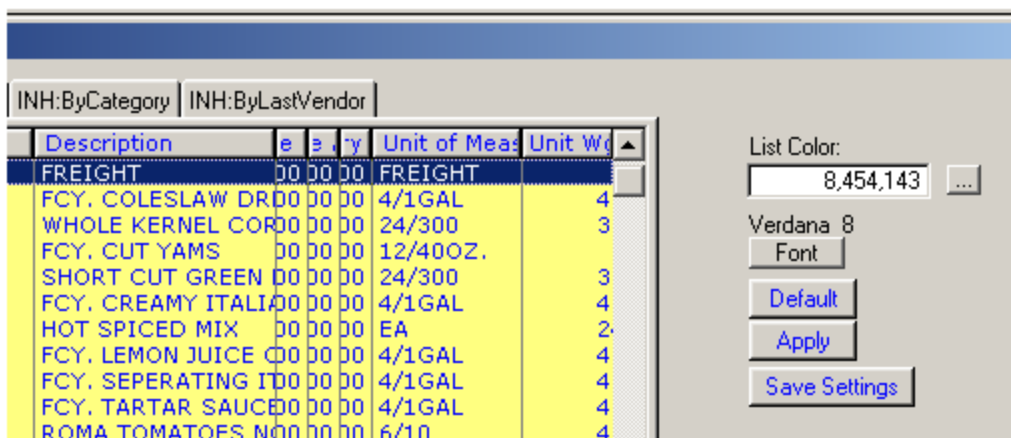


Values are stored in an inifile and applied when starting the window.
Buttons call up the standard Windows Color Dialogue and Font Dialogue.
Default will load the original design fonts and colors and save them to the ini.
Apply will apply the current selections to the listbox, for the current session, but not save them.
Save Settings will save the current settings to the ini for this and further uses of the listbox.
You can delete the color control or font control if you only want to give the user partial functions.
You can, of course move the controls individually as needed to position.
The only prompts you need to fill are which listbox to control, and the ini file to use. It defaults to BST.INI so as not to affect the main ini for the app.
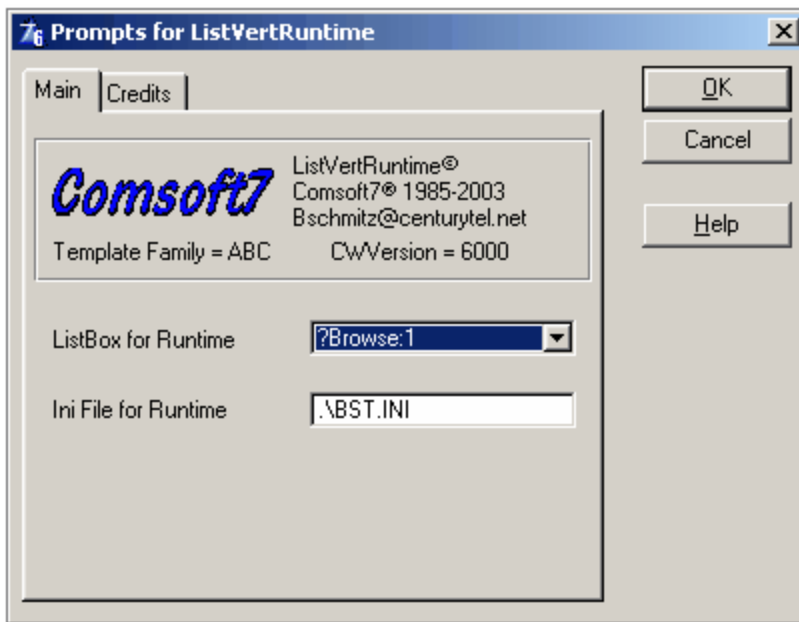
## 2.8    LookupButton

The LookupButton provides a button and template to call a 'select enabled' procedure to fill in form fields.
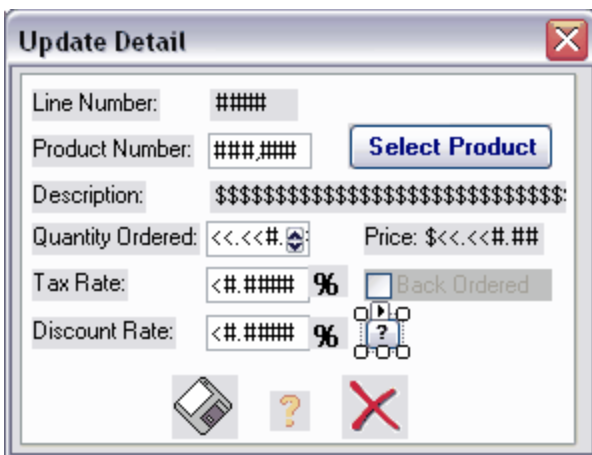


The button is small with the VCR Question Mark Icon

Tell the template which field to fill, and the lookup procedure and Field with the data to place in the lookup field.
Add any procedure parameters for the Select procedure if needed.
If you have addition fields to fill, use the Insert button to add them to the list.



## 2.9    PerCentGraph

PerCentGraph is a control template that uses an image control to create a horizontal bar type graph. PerCentGraph adds a group, Panel, and image control in a horizontal layout shown in below window in the ide. Strings for clarity added by the developer.
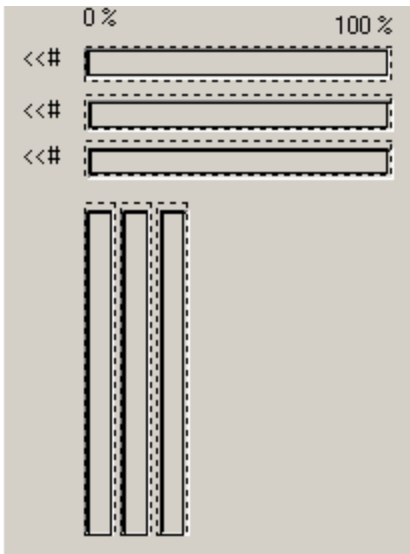
Right click on the control and select the actions tab for the prompts.
You only need to select the Image to use for the bar graph and the variable to represent with the graph.
Any Image format supported by the Clarion IDe can be used, and the Variable should be a numerical value representing a percentile in the range of 1-100, ie 65(%) The length of the image is 100 and {PROP:WIDTH} is set to the variable.
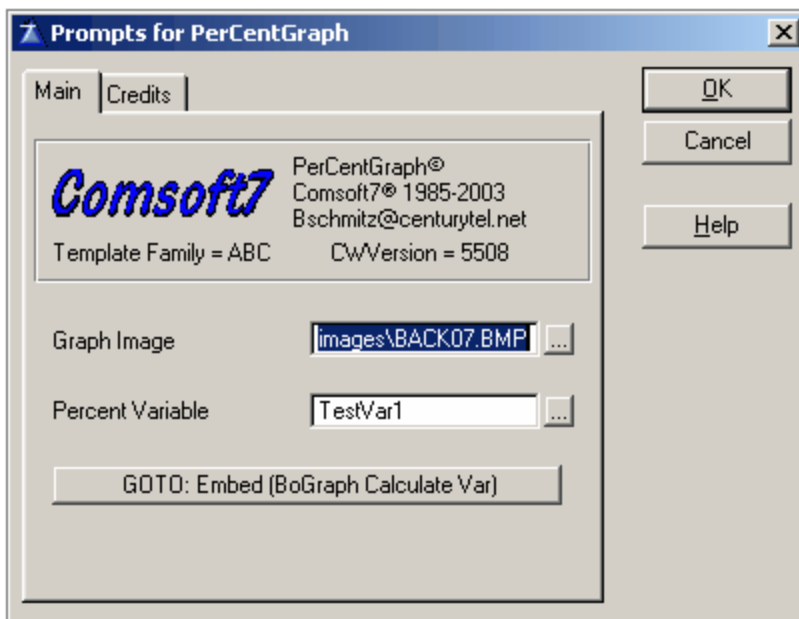


A Goto button is provided to go to an embed in the routine before the variable is used and assigned to the graph for any necessary hand coding.
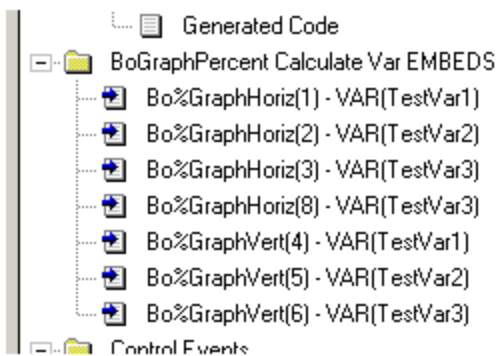
Below is screen shot of Horizontal bar graph in action.
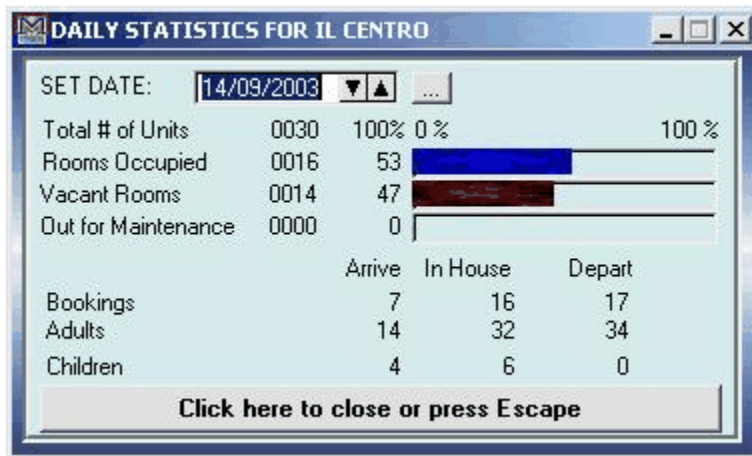


## 2.10    PerCentGraphVert

PerCentGraphVert control template that uses an image control to create a Vertical bar type graph. PerCentGraphVert adds a group, Panel, and image control in a Vertical layout shown in below window in the ide.
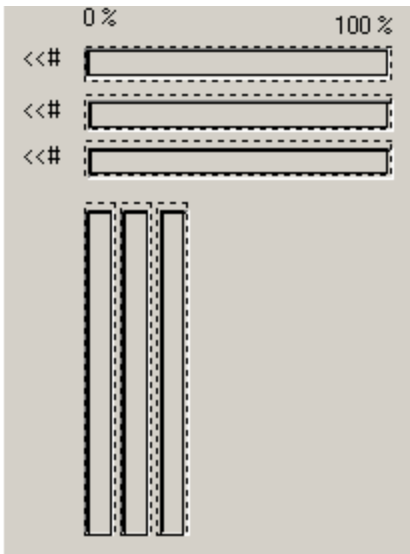
Right click on the control and select the actions tab for the prompts.
You only need to select the Image to use for the bar graph and the variable to represent with the graph.
Any Image format supported by the Clarion IDe can be used, and the Variable should be a numerical value representing a percentile in the range of 1-100, ie 65(%) The Height of the image is 100 and {PROP:HEIGHT} is set to the variable.
The image is then repositioned to start at the bottom of the panel.
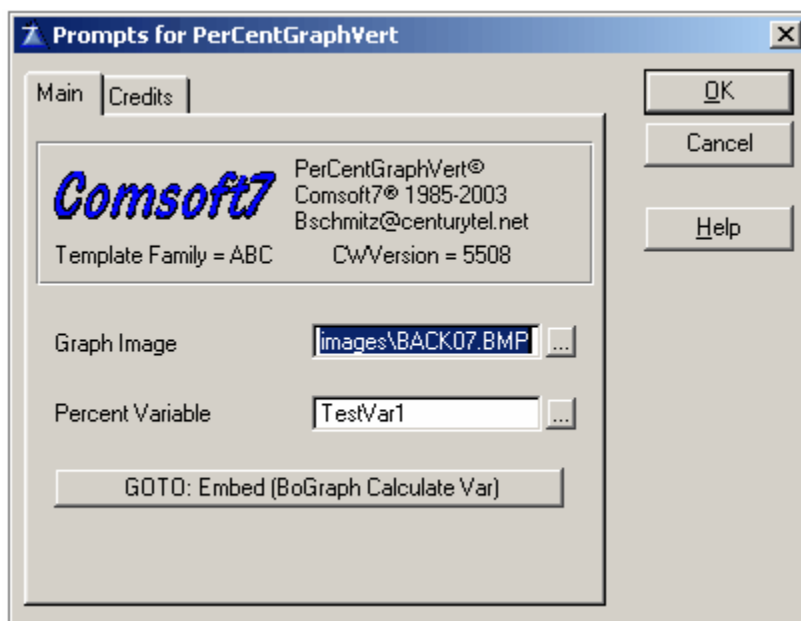


A Goto button is provided to go to an embed in the routine before the variable is used and assigned to the graph for any necessary hand coding.

```
   └─ 🗒 Generated Code
─ 📁 BoGraphPercent Calculate Var EMBEDS
   ├─ 🔁 Bo%GraphHoriz(1) - VAR(TestVar1)
   ├─ 🔁 Bo%GraphHoriz(2) - VAR(TestVar2)
   ├─ 🔁 Bo%GraphHoriz(3) - VAR(TestVar3)
   ├─ 🔁 Bo%GraphHoriz(8) - VAR(TestVar3)
   ├─ 🔁 Bo%GraphVert(4) - VAR(TestVar1)
   ├─ 🔁 Bo%GraphVert(5) - VAR(TestVar2)
   └─ 🔁 Bo%GraphVert(6) - VAR(TestVar3)
─ 📁 Control Events
```
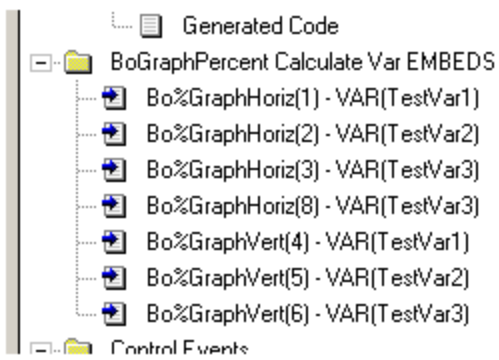
Runtime display of vertical graph.
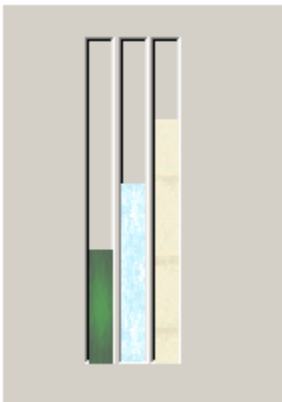
# 3    Extension Templates
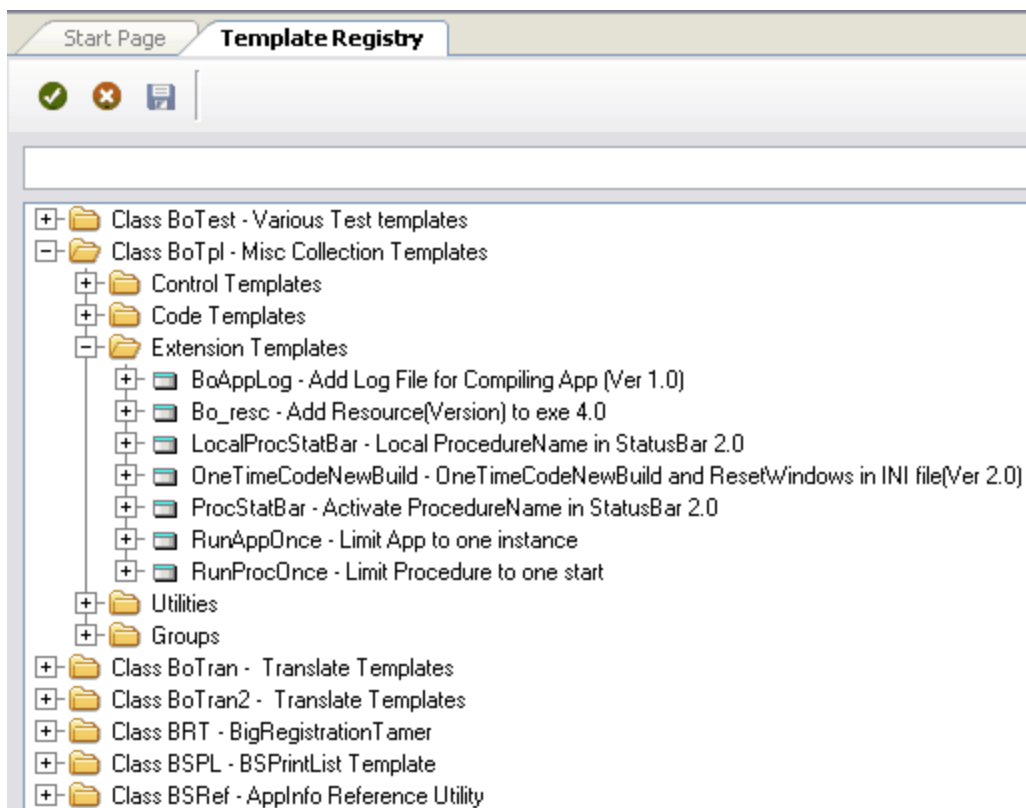


## 3.1    BoAppLog

**BoAppLog** adds the ability to create a log file with your application logging **dictionary changed**, and **procedures changed** with each compile. Gives you the ability to **add one-time comments** to the log file as well.  It does this last after a compile, and does not affect your app in any adverse way.

BoAppLog tab / Credits tab

BoAppLog
Comsoft7® 1985-2009
bo@comsoft7.com

Template Family = ABC     CWVersion = 7000

When active, BoAppLog will maintain a log file for your application indicating changes in each compile or build. Records if dictionary changed and which procedures changed. You can add one-time comments to the log file for each particular build.
Will not affect app in any way.

☑ Log Compiles -- Date/Time/Changes
Log File:     invoiceBoAppLog.Log
☑ Add one time comments (when changed)

Major appearance changes on all forms.

Example log entries


****** BSRef.EXE Compiled on  8/13/2009 at  8:12 AM
Procedure -  FilesUsed  - Changed on  8/13/2009 at  8:10 AM
Procedure -  Files_Keys  - Changed on  8/13/2009 at  8:11 AM
Procedure -  FilesAll  - Changed on  8/13/2009 at  8:11 AM
------
Moved buttons to make for consistent appearance.
****** BSRef.EXE Compiled on  8/13/2009 at  8:15 AM  ******
****** BSRef.EXE Compiled on  8/13/2009 at 12:12 PM  ******
Dictionary Changed on  8/13/2009 at 11:38 AM -- E:\BoDev\BSRef.dct
Procedure -  Main  - Changed on  8/13/2009 at 12:06 PM
Procedure -  MainProject  - Changed on  8/13/2009 at 12:07 PM
Procedure -  PWREPORT  - Changed on  8/13/2009 at 12:11 PM
Procedure -  PWPReport  - Changed on  8/13/2009 at 12:12 PM
Procedure -  BRTWin  - Changed on  8/13/2009 at 11:46 AM
Procedure -  RegApp  - Changed on  8/13/2009 at 11:48 AM
****** BSRef.EXE Compiled on  8/13/2009 at 12:24 PM  ******
Procedure -  Main  - Changed on  8/13/2009 at 12:24 PM


## 3.2    Bo_resc

Bo_resc is a global Extension template that adds Version Information Resources to the exe or dll. It includes an auto increment build feature for each compile.

The Build Number is stored in %ApplicationBld.txt. It interacts with the **AppSpecsControl**. 32bit only. The defaults are set up for my use, and you can change them in the template editor by editing the DEFAULT Settings, Example --- #PROMPT ('Copyright', @s255), % ResLegalCopyright, DEFAULT('Copyright<169> Comsoft7<174>1985-2009')

Properties (32-bit) | From | Credits

*Comsoft7*
Bo_resc
Comsoft7® 1985-2009
bo@comsoft7.com
Template Family = ABC          CWVersion = 7000

☑ Add file properties to the invoice.exe file
☑ Enable Special Build Autoinc

Resource File Information

File Version          1 . 0 . 0 . 0
Description           Invoice Program
Copyright             Comsoft7©1985-2008
Company Name          Comsoft7®
Internal Name         invoice.app
Original Filename     invoice.exe
Product Name          Invoice Program

Following are Auto Generated
Product Version       CW6300  ABC
Product Build         1

Add Custom Resources

[ Insert ] [ Properties ] [ Delete ] [↑] [↓]

To Paste              (tm)  ®  (c)  ©
[ Reset To Paste ]

The File Version defaults to 1.0.0.0, salt to taste. Up to 99.99.99.99<g>.
Now is working with W2K and XP.

Fill in the Description and Product Name.

Most of the prompts are calculated for you, and you can change the defaults in the template

to represent your company in the prompts.

#PROMPT ('Copyright', @s255), %ResLegalCopyright, DEFAULT('Comsoft7<169>1985-2003')
#PROMPT ('Company Name', @s255), %ResCompanyName, DEFAULT('Comsoft7<174>')

You can enable the Special build autoinc to increment the build each time you compile. The Build Number is stored in %ApplicationBld.txt in your compile
directory, or whereever your Redirection file may take *.txt files. You can delete the file to start over, or edit it with notepad to change the version number. It stores
the last used number and increments it on compile.

Following is the results on XP.
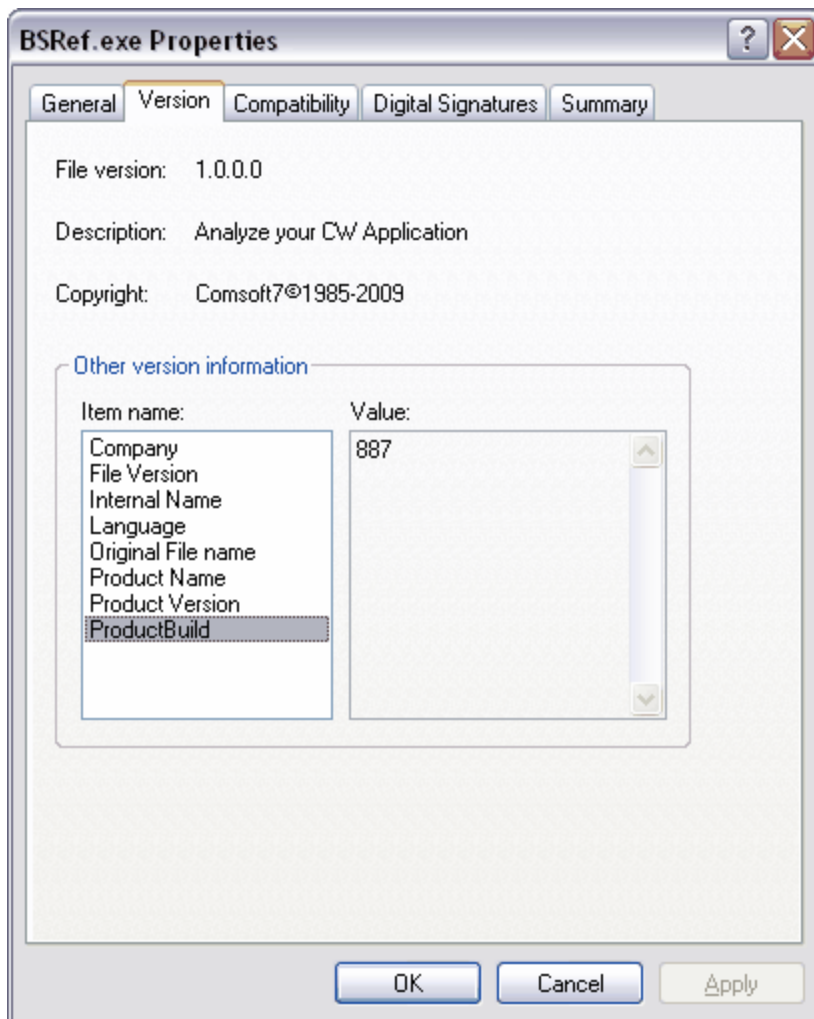


With added custom resources.
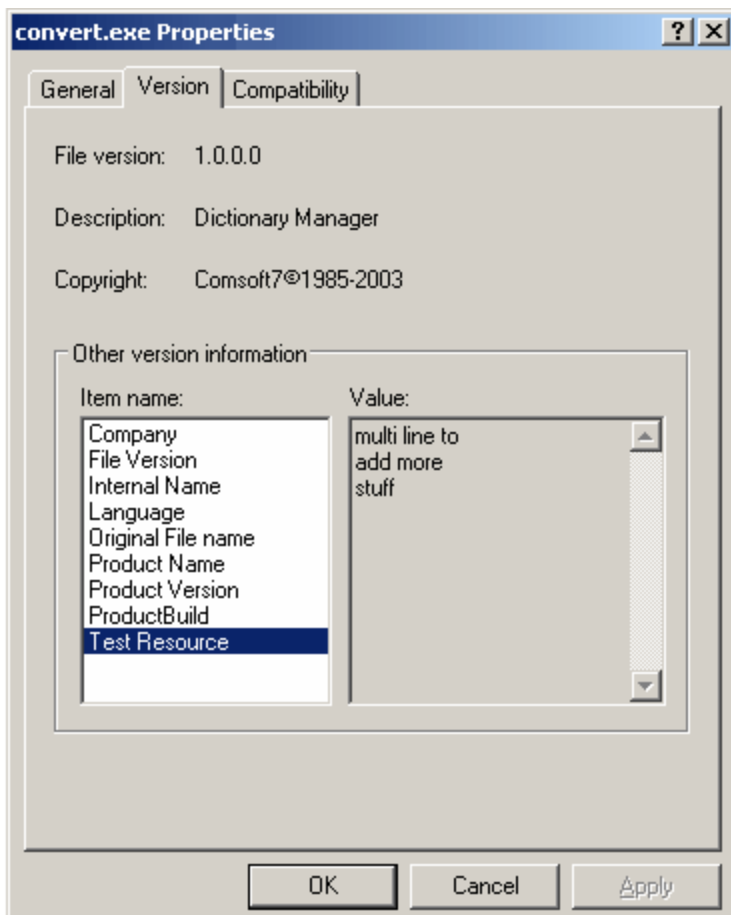
## 3.3   ProcStatBar

**ProcStatBar** is a Global Extension, and **LocalProcStatBar** is the local extensions that work together to put the name of the active procedure and template type
into the status bar of the main frame.
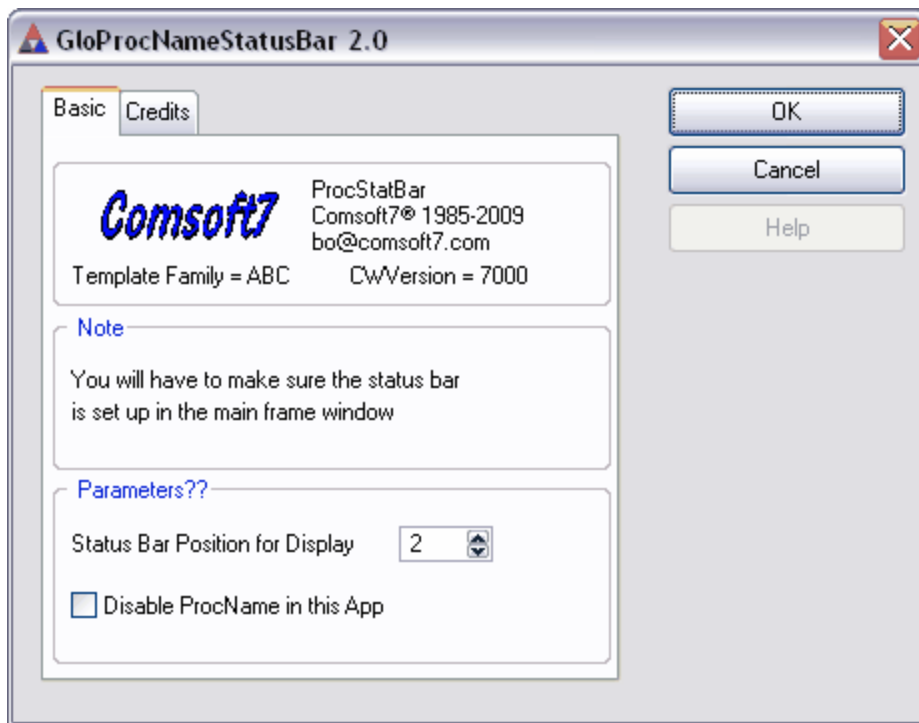
Remember to set up the status bar in the frame, and just tell the template which Position you want to display the Procedure name in when the app is running. It calculates the display name from the Procedure name.

**Application**

File   Edit   Browse   Window   Help

Labs    Labs2   Info

**Browse the LABELS File**

LAB:TYPE

| Label Type | Page Width<br>Page Height | Label Width<br>Label Height | Top Margin<br>Left Margin | Font<br>Font Style<br>Font Size | Bar Code Font<br>Bar Code Font Style<br>Bar Code Font Size |
|---|---|---|---|---|---|
| .1 x 2-5/8 Cur | 8.50<br>10.50 | 2.62<br>1.00 | 0.62<br>0.50 | Arial<br>700<br>     12 | 3 of 9 Barcode<br>700<br>    24 |
| .Avery 2-Up | 8.50<br>11.00 | 4.00<br>1.50 | 0.50<br>0.50 | Arial<br>400<br>    10 | 3 of 9 Barcode<br>700<br>    24 |
| .Avery 3-Up5351 | 8.50<br>10.50 | 2.63<br>1.00 | 0.62<br>0.50 | Arial<br>400<br>    8 | 3 of 9 Barcode<br>700<br>    20 |
| .Avery 4-Up | 8.50<br>11.00 | 2.00<br>1.00 | 0.25<br>0.25 | Arial<br>400<br>    8 | 3 of 9 Barcode<br>700<br>    18 |
| .Avery 8163 2x5 | 8.50<br>11.00 | 4.00<br>2.00 | 0.75<br>0.50 | Arial<br>700<br>    16 | 3 of 9 Barcode<br>700<br>    24 |
| .Avery 8164 2x3 | 8.50<br>11.00 | 4.00<br>3.33 | 1.50<br>0.50 | Arial<br>400<br>    20 | 3 of 9 Barcode<br>700<br>    24 |

Browsing Records       BrowseLABELS - Browse

### 3.3.1   LocalProcStatBar

**ProcStatBar** is a Global Extension, and **LocalProcStatBar** is the local extensions that work together to put the name of the active procedure and template type
into the status bar of the main frame.
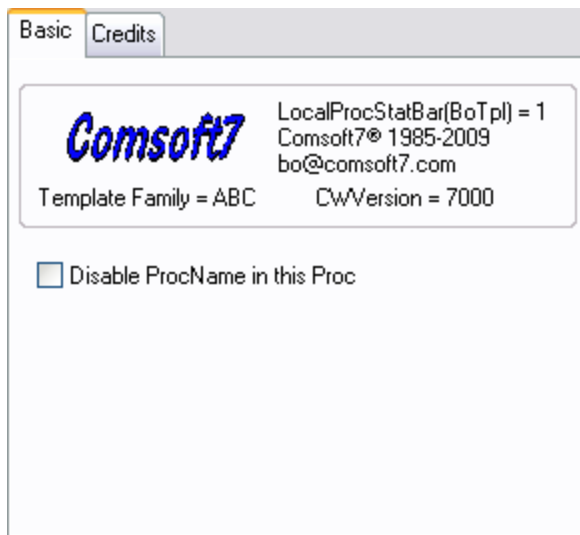
You can disable the Procedure Name in any procedure you want.

## 3.4   OneTimeCodeNewBuild

**OneTimeCodeNewBuild** is an extension template for the frame that runs once on a new install or on an updated version of the program. Uses Ini Manager (ABC only) and has an embed point for code to run once. Has a checkbox if you want to reset all the window sizes and positions stored in the Ini file. (If you redesigned the screens.) It is activated by manually incrementing or changing a version value. The version value is stored in the App Ini file under ('RWSZ','RWSZFlag').  It works in conjunction with the **ResetWindows** Code Template which can be put under a button and will reset the windows at runtime at an application restart.

Simply change the value stored in Version or Build to trigger the one time event on the first time the application is run, on install, or on an update to the application. You can go directly to the embed point to enter code to execute. This code can be changed on each release if needed. The checkbox will loop thru and reset all the stored sizes of windows in the ini file if you have redesigned the screens.

It stores the version or build value in the ini file, so deleting the file or changing the value of ('RWSZ','RWSZFlag') will also cause the code to fire.

## 3.4.1   Code Templates

### 3.4.1.1 ResetWindows

**ResetWindows** is a code template that can be placed under a reset button on the main frame, that will cause the **OneTimeCodeNewBuild** template to fire its embed point on startup of the app, repositioning and resetting the sizes on all windows in the app. This can be used if the end user has lost a window.<seg>
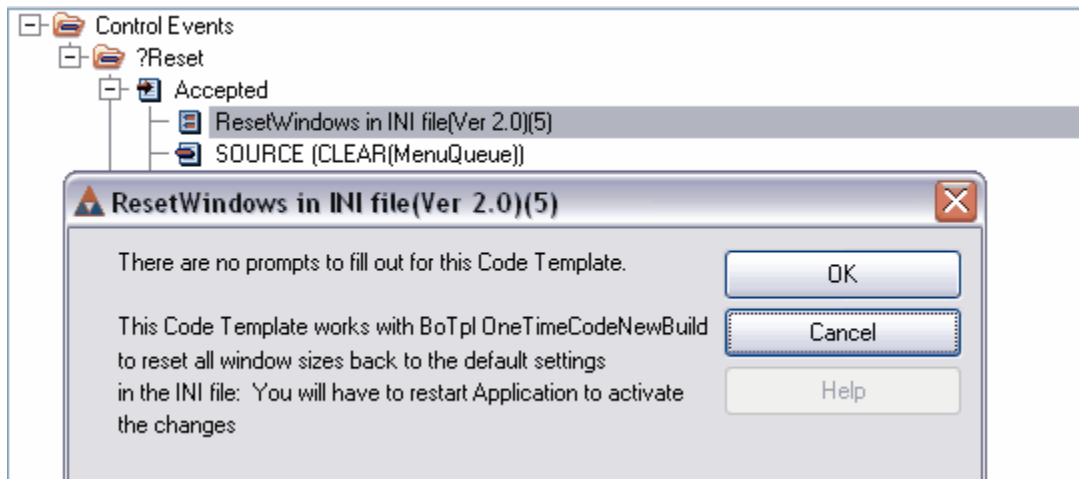


## 3.5    RunAppOnce

**RunAppOnce** is a Global Extension using Mutex to allow only one instance of your application to run.



**Get App Name** will extract the name of the Frame or First Procedure which is what the Windows API 'FindWindowA' uses.

## 3.6 RunProcOnce

Add this extension template to limit any procedure to one instance.
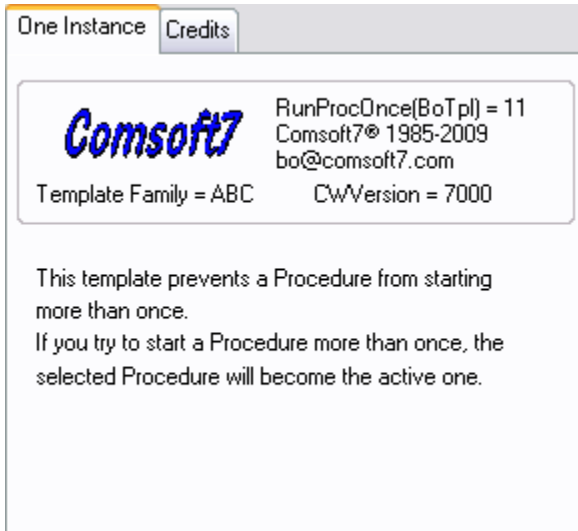


## 4 Utility Templates
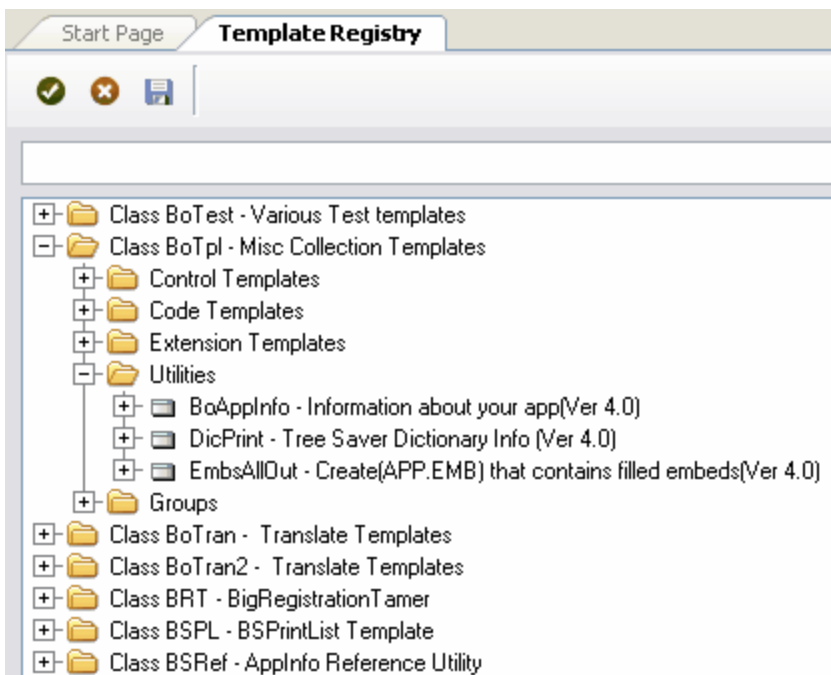


## 4.1 BoAppInfo

BoAppInfo is a utility to create a text file and/or print information about your app. Now includes module data and does an instring search of the txa for all
supported image files, lib references, and wav's in your app. Lists all the usual stuff and includes formulas.

You find it from the menu with Application\Template Utility or hot keys Ctrl+U
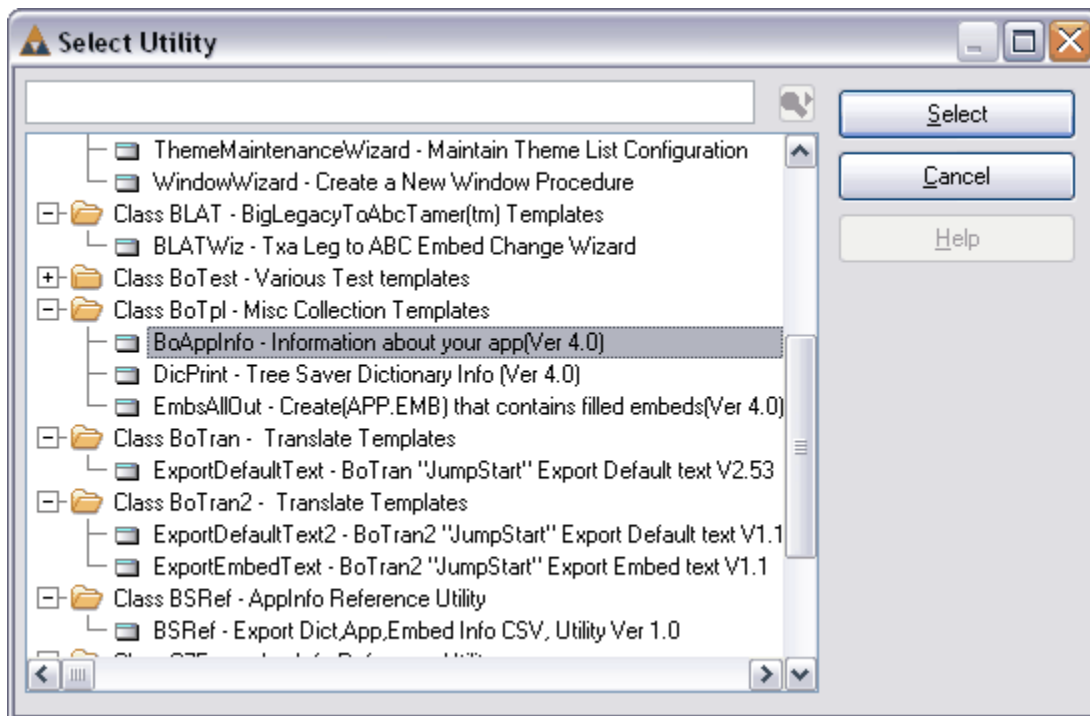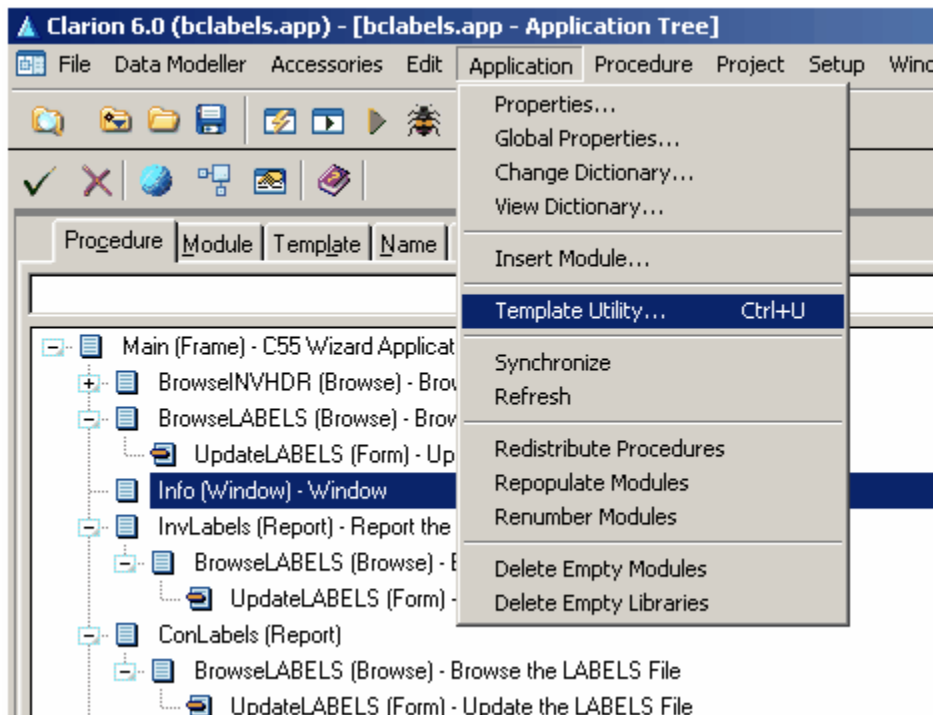




You then have a zillion options that you can select from.<g>
You can use the default filename, or select your own.

There are three buttons to quick select the most common options.
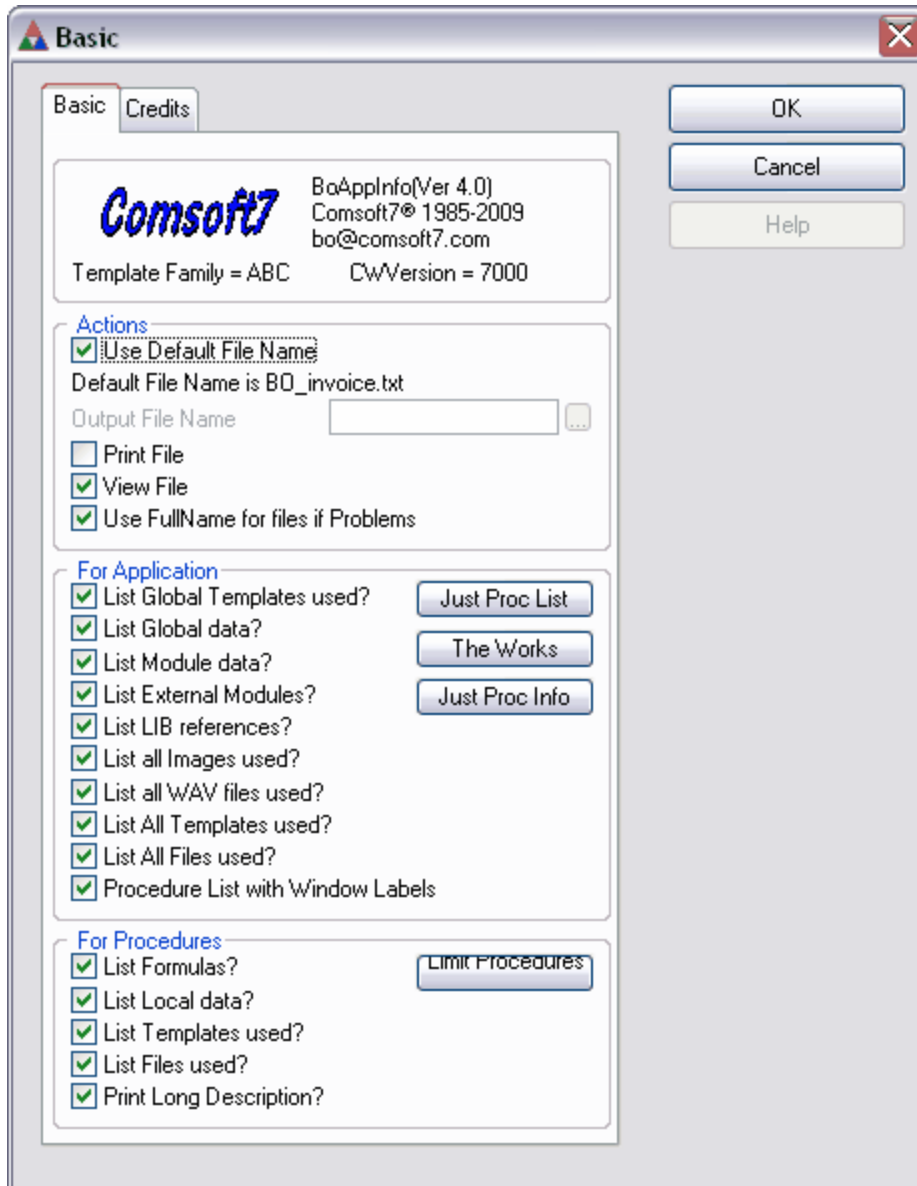


The limit procedures buttons brings up addition prompts to further limit the output to the ones you want.
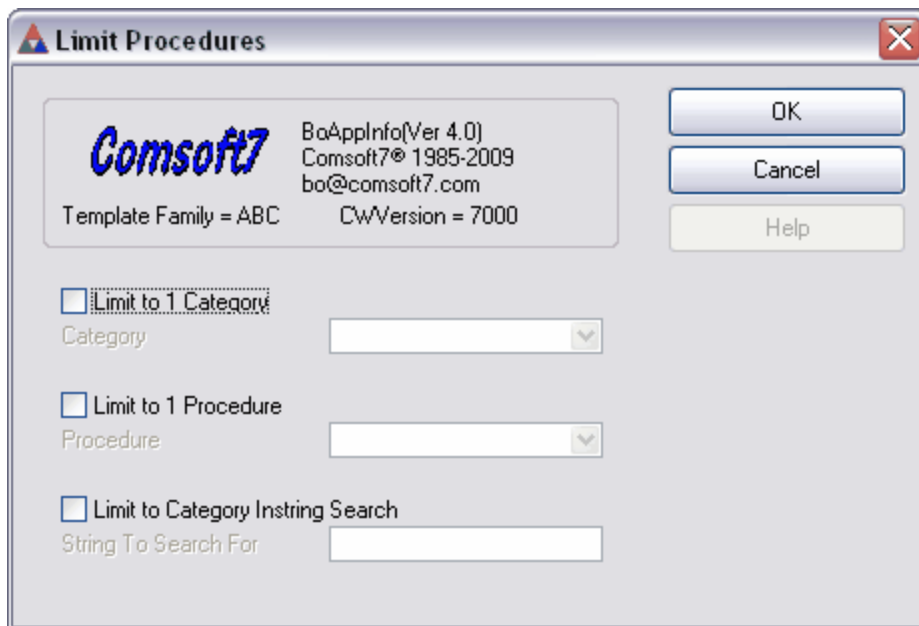
Further options are available if you select a limiting feature.
The Category Instring search was added to limit to Key words or changes that you can
record in the procedure per request of Greg Berthume. It does an instring
search of %ProcedureCategory and %ProcedureTemplate for your string.

## 4.2    DicPrint

DicPrint is another dictionary to text/print utility. This one is very concise, and you can
configure it to include keys and relationships if you want them. It has the
ability to select all files or just the ones you want.
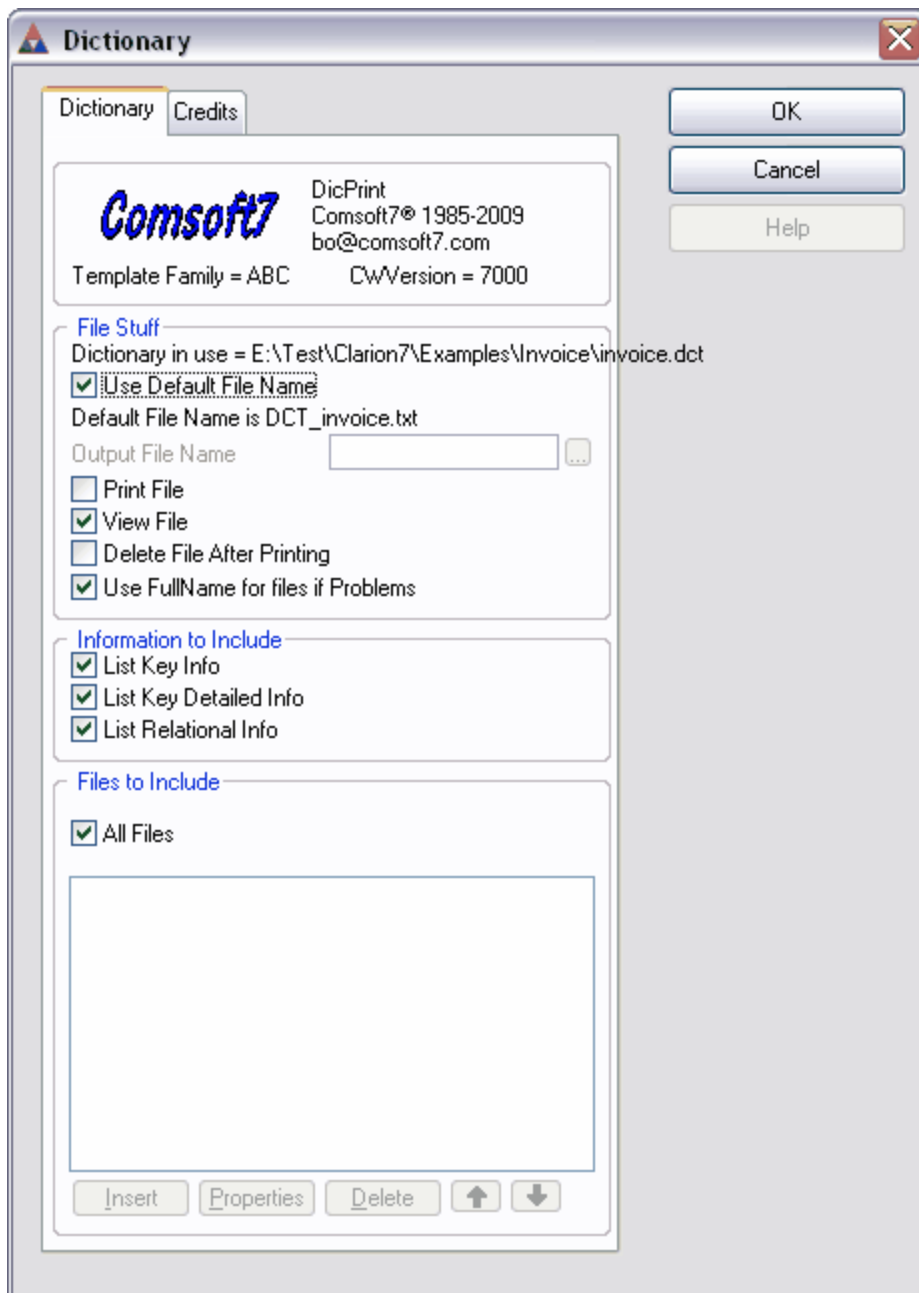
Just uncheck the all files, and you can then select one or as many files as you want included in the output file.

## 4.3    EmbsAllOut

EmbsAllOut is an utility template to capture global, module, and procedure embedded code and outputs them to a text file in a format suitable to copying to templates or just cutting and pasting the code in other embeds.
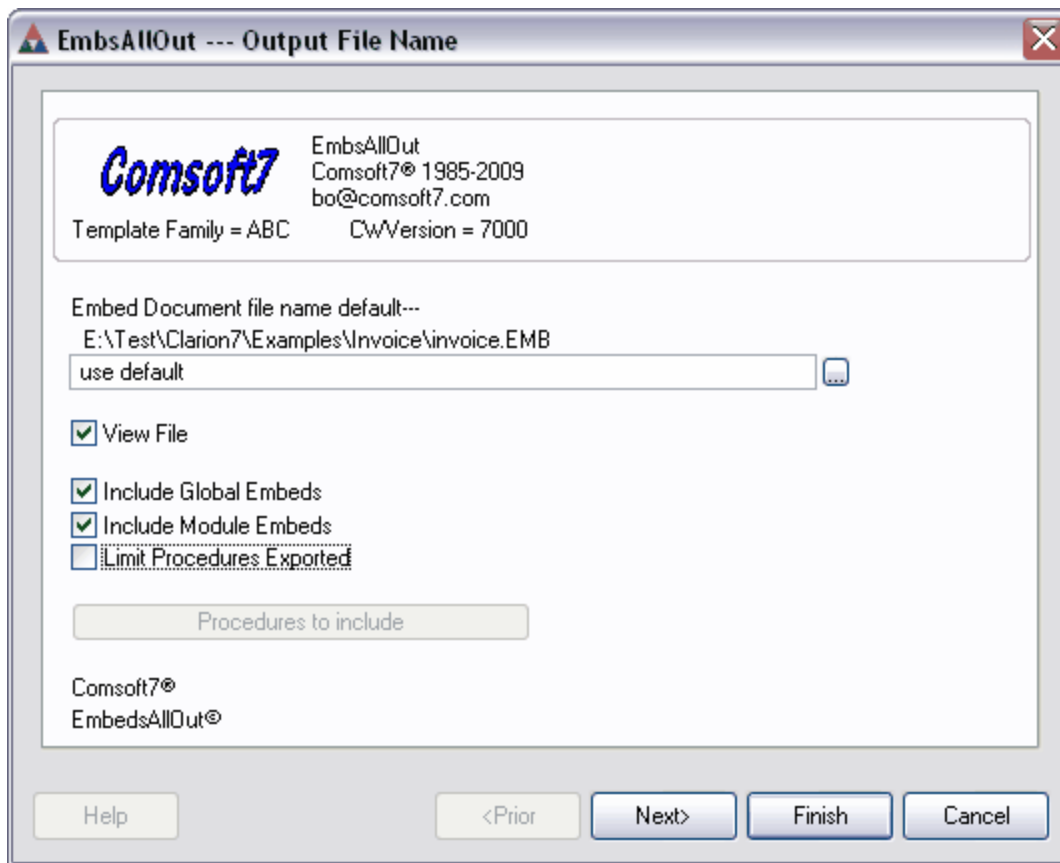
EmbsAllOut defaults to outputting all embeds to a file named %application.emb. If you are happy with the default output name, hit finish, and it creates the output
file for you. You can turn off outputting Global and Module embeds by unchecking the boxes. If you want to limit the procedures exported, check the box and hit
the Procedures to Include Button.
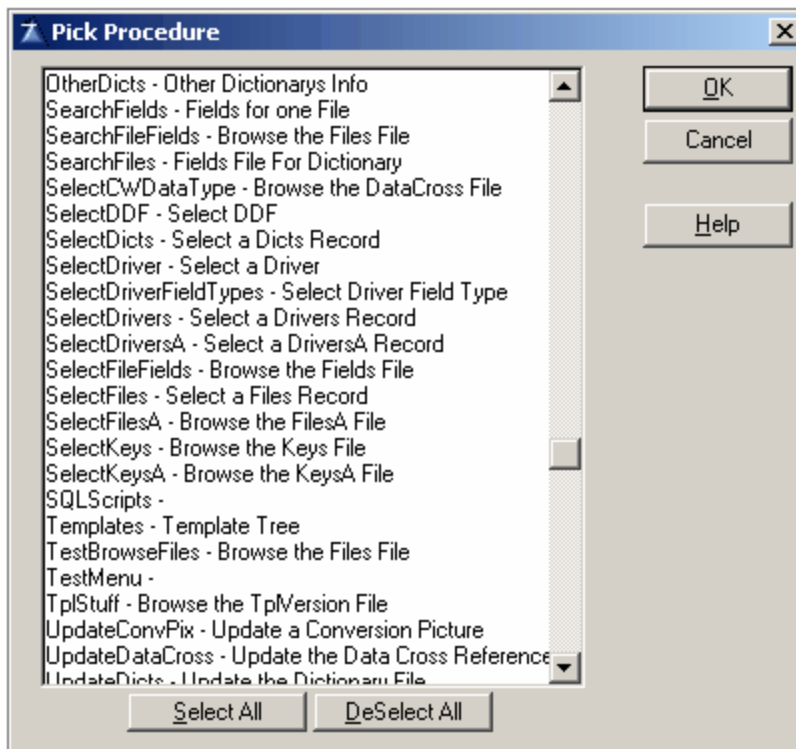
# 5    Label Stuff Included

A labels file and procedures are included in the Examples directory under 3rdParty
\Examples\BoTpl that work well with the Barcode Label and Mailing label
templates.

The file definition is stored in a txd file for import into your dictionary (LabelFile.txd), or you
can just import the file itself.
The browse Label and update form are as follows, and are stored in a c55 abc txa
(labelprocabc.txa) for import.
The included Labels tps file has about 77 or so labels pre-entered. The fonts, styles and
sizes can be changed as required, and the update form has font lookup buttons.

## Browse the LABELS File

| Label Type | Page Width / Page Height | Label Width / Label Height | Top Margin / Left Margin | Font / Font Style / Font Size | Bar Code Font / Bar Code Font Style / Bar Code Font Size |
|---|---|---|---|---|---|
| .1 x 2-5/8 Cur | 8.50 / 10.50 | 2.62 / 1.00 | 0.62 / 0.50 | Arial / 700 / 12 | 3 of 9 Barcode / 700 / 24 |
| .Avery 2-Up | 8.50 / 11.00 | 4.00 / 1.50 | 0.50 / 0.50 | Arial / 400 / 10 | 3 of 9 Barcode / 700 / 24 |
| .Avery 3-Up5351 | 8.50 / 10.50 | 2.63 / 1.00 | 0.62 / 0.50 | Arial / 400 / 8 | 3 of 9 Barcode / 700 / 20 |
| .Avery 4-Up | 8.50 / 11.00 | 2.00 / 1.00 | 0.25 / 0.25 | Arial / 400 / 8 | 3 of 9 Barcode / 700 / 18 |
| .Avery 8163 2x5 | 8.50 / 11.00 | 4.00 / 2.00 | 0.75 / 0.50 | Arial / 700 / 16 | 3 of 9 Barcode / 700 / 24 |
| .Avery 8164 2x3 | 8.50 / 11.00 | 4.00 / 3.33 | 1.50 / 0.50 | Arial / 400 / 20 | 3 of 9 Barcode / 700 / 24 |

LAB:TYPE

Insert    Change    Delete

Close    Help

## Changing a LABELS Record

General

Label Type: .Avery 4-Up

Page Width: 8.50    Top Margin: 0.25    Label Width: 2.00

Page Height: 11.00    Left Margin: 0.25    Label Height: 1.00

Font: Arial     Bar Code Font: 3 of 9 Barcode

Font Style: 400     Bar Code Font Style: 700

Font Size: 8     Bar Code Font Size: 18

OK    Cancel    Help

# 6    Obsolete Label Procedure

If you have been using BoTpl since C4 days, you may have been using the old procedure and extension template first used for the Labels templates.  You can still use them by uncommenting the following line in Bo.Tpl

```
#!-------------------------------------------------------
#!-------------------------------------------------------
#TEMPLATE (BoTpl, 'Misc Collection Templates'),Family('cw20'),Family('abc')
#!-------------------------------------------------------
#!-------------------------------------------------------
#INCLUDE ('BoGrp.tpw')       #! extra groups stuff
#INCLUDE ('Beaot.tpw')       #! Util-to export all embeds in app
#INCLUDE ('Bo_resc.tpw')     #! App Ext Add Version Info to exe and dll
#INCLUDE ('boappinfo.TPW')   #! Util-App Data,templates,images,formulas,files
#INCLUDE ('BoAppLog.tpw')    #! Add a compile Log to your application.
#INCLUDE ('Bodicprt.TPw')    #! Util-Dict to textfile and print utility,concise
#INCLUDE ('BoDLR.tpw')       #! Dynamic list runtime controls
#!INCLUDE ('bolab.TPW')      #! Obsolete Report proc and AMLabel Extension templates
<<<<<<<<<<<<<<
#INCLUDE ('BoLabCon.tpw')    #! Label control templates
#INCLUDE ('BoPrCent.tpw')    #! Graphic Percent bar using image
#INCLUDE ('boprocn.tpw')     #! GloExt Display Proc name in Status bar
#INCLUDE ('BoSearch.tpw')    #! Instring Search of file records and Memos
#INCLUDE ('BoVerInfo.tpw')   #! AppSpecsControl app info into about screen
#INCLUDE ('CopyButton.tpw')  #! ABC Browse Copy record and children button.
#INCLUDE ('RunOnce.tpw')     #! Run App Once using Mutex, run procedure once.
#INCLUDE ('RWSZ.tpw')        #! Frame Ext onetime code to execute on new version
#!
```