



**Bo's Template
Tamers**

By Bo Schmitz

© <1985-2004> ... Comsoft7

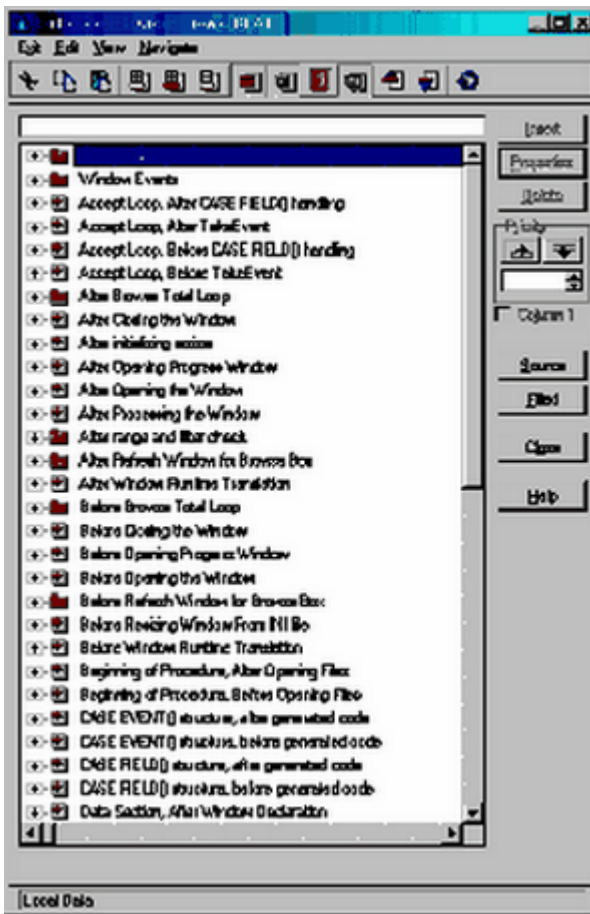
Table of Contents

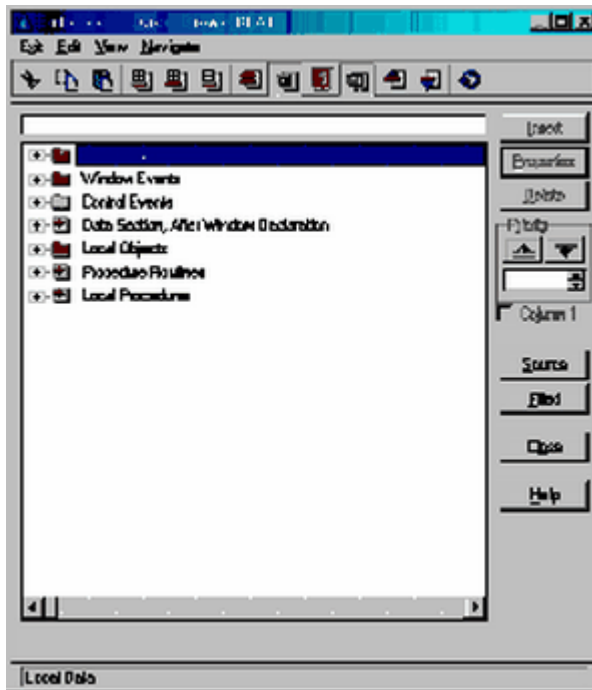
Foreword	0
Part I BLAT	2
1 Whats New	5
2 How To Use	6
3 What BLAT Does	10
Blat.exe	12
4 What Is BLAT	13
5 Contact Us	17
Index	0

1 BLAT

BigLegacyToABCTamer Template(tm) AKA (BLAT)

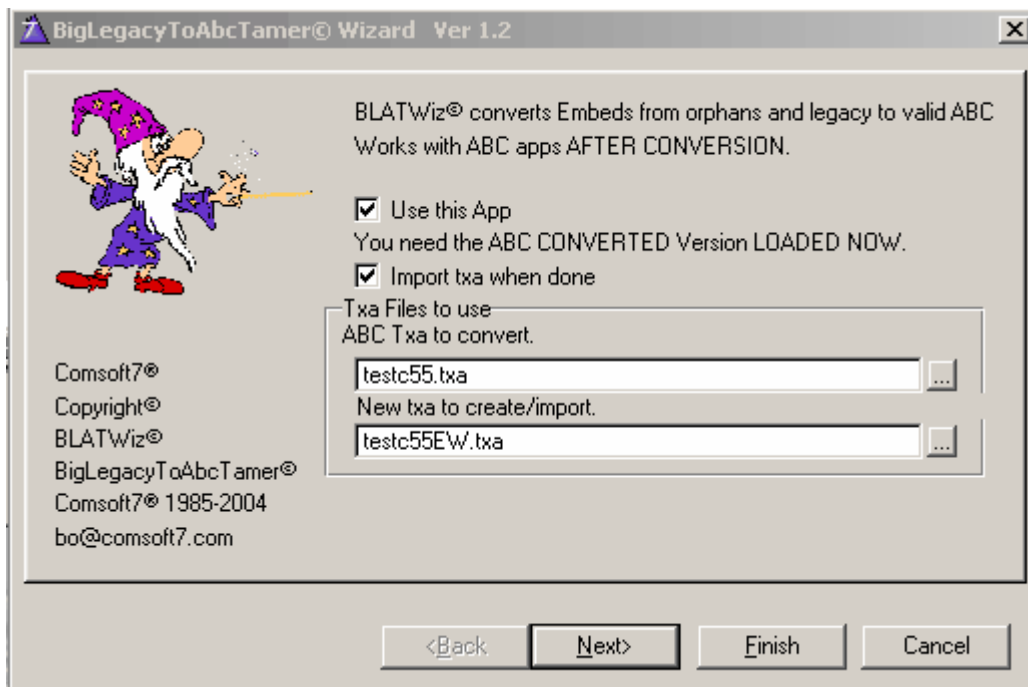
Introducing BLATWiz© Template



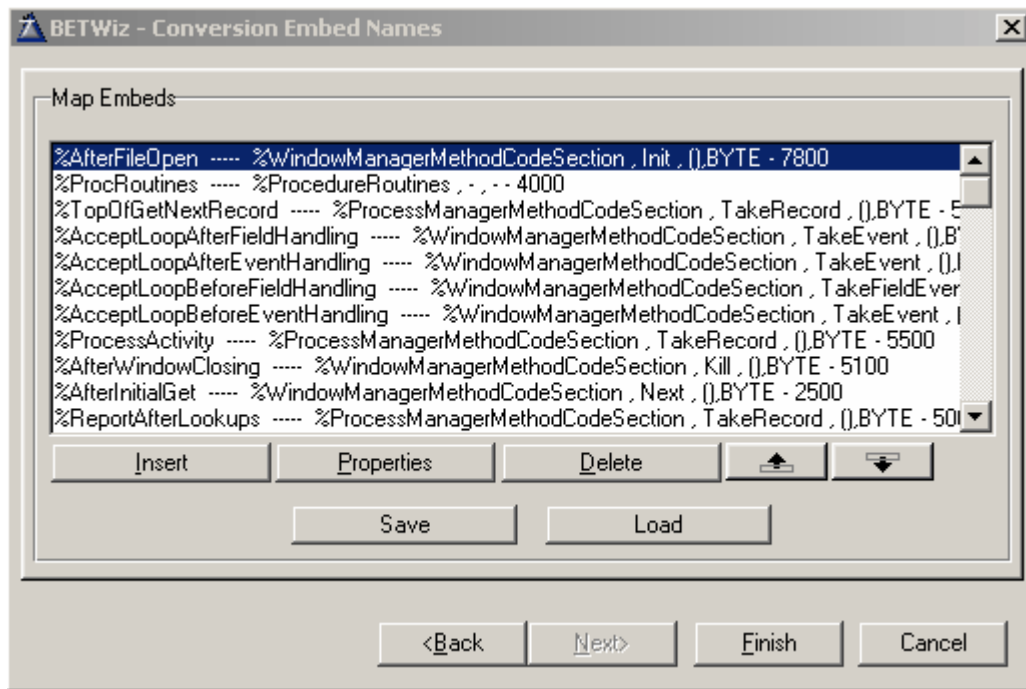


Turn This Into This

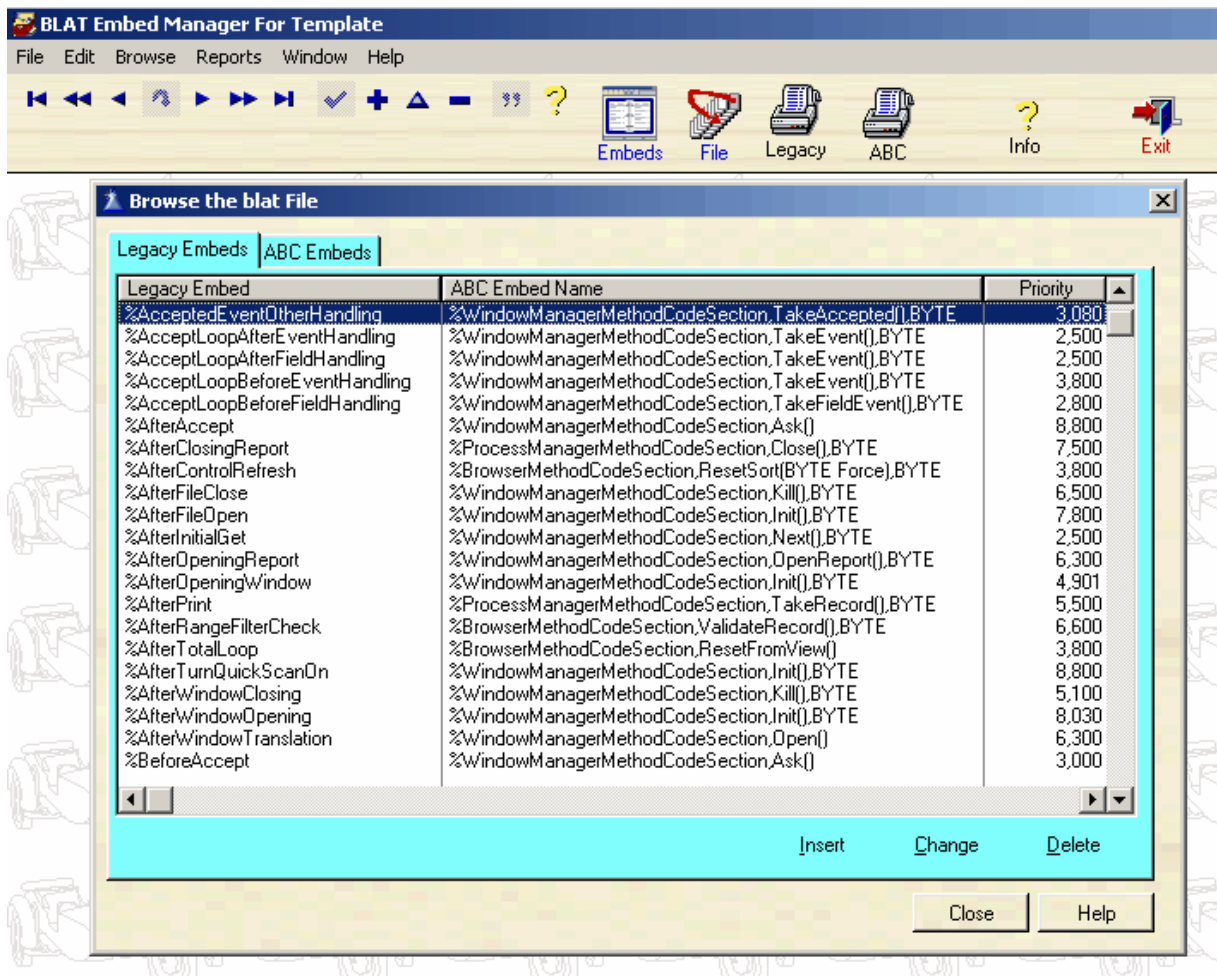
The above screenshots show an example of the embeditor in collapsed filled embeds Mode before and after running BLATWiz© Template



The embeds it uses to Convert the app.



Edit embeds here if desired or use manager Blat.exe.
You can save or load the embeds from this screen and use them in Blat.exe.



1.1 Whats New

November 10, 2004 BLAT Ver 1.3

Bug fix, on "priorities" numbers for converting legacy embeds '%ControlPostEventHandling' and '%ControlPreEventHandling'. They were retaining original priorities, regression from Ver 1.2 fix.

Feb 10, 2004 BLAT ver 1.2

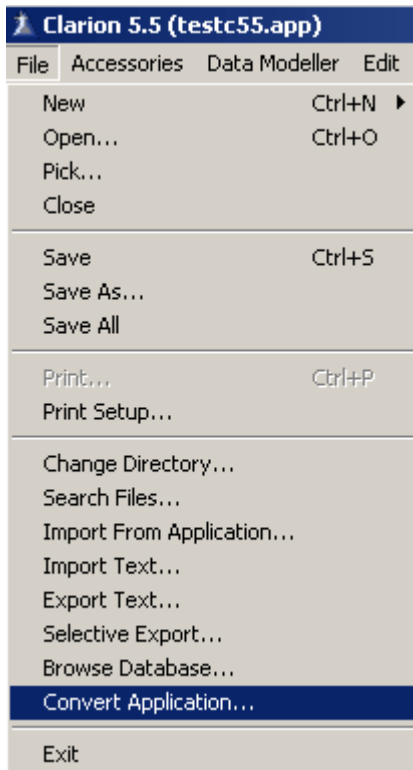
Bug fix, BLAT would incorrectly handle "Call a Procedure" code embed under some circumstances. This would result in errors to the new txa file being created.

1.2 How To Use

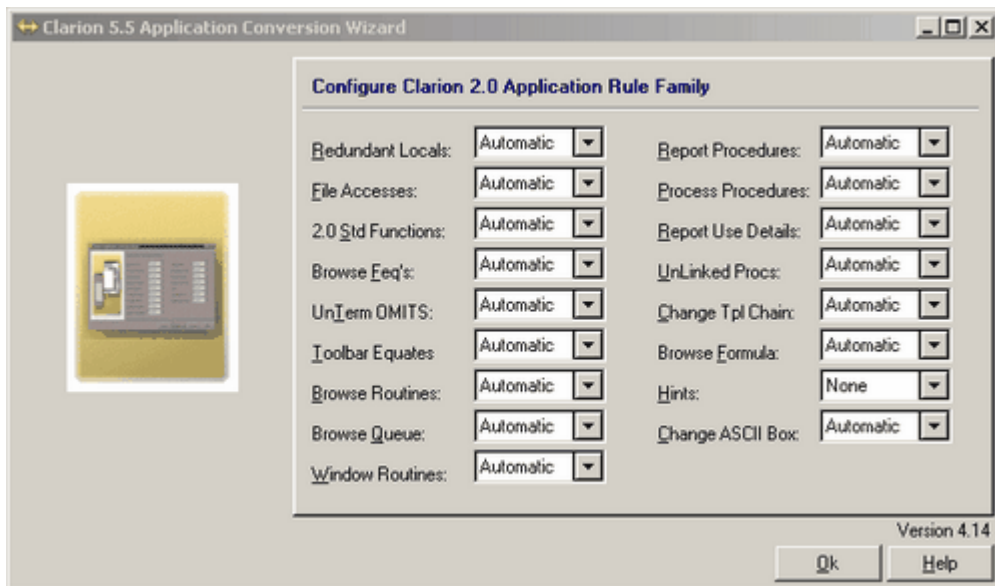
BLAT is a Utility Wizard. It is only available if an ABC app is loaded.

Steps to use.

1. Convert the Clarion (Legacy) app to ABC using the IDE Conversion Utility.



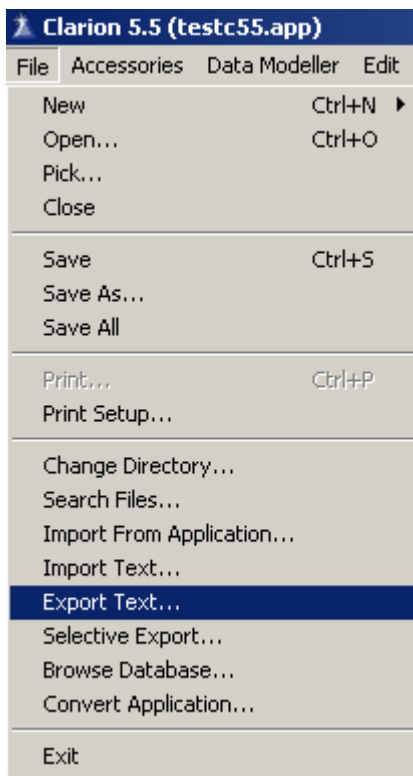
Selecting Automatic for all settings in the conversion process for CW2 templates seems to have good results.



2. Load and try to compile the newly created ABC app.

If it does not compile, you must repair any items necessary to get it to compile before running BLAT.

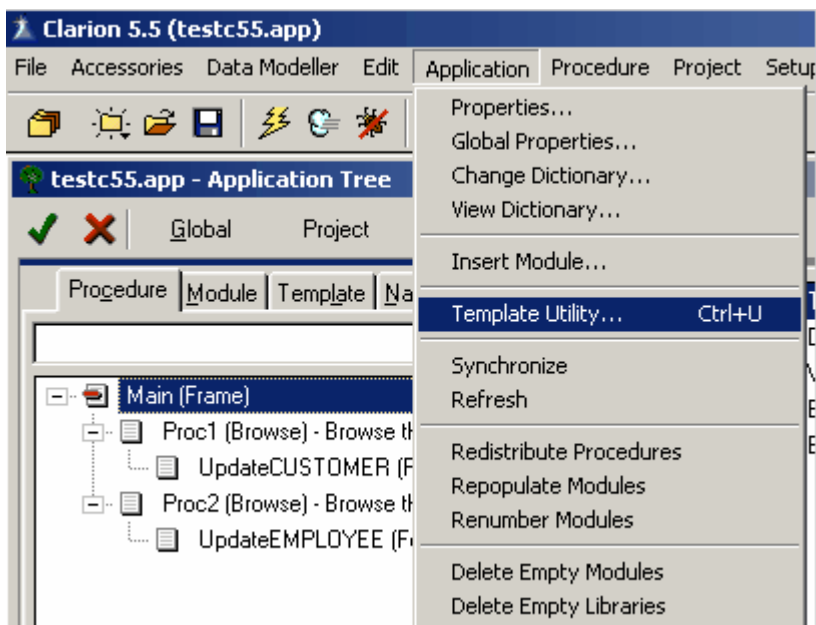
3. Once it compiles, export the app to a "TXA" You must do this manually, because there is a bug in the #EXPORT template command that does not give a good txa file.

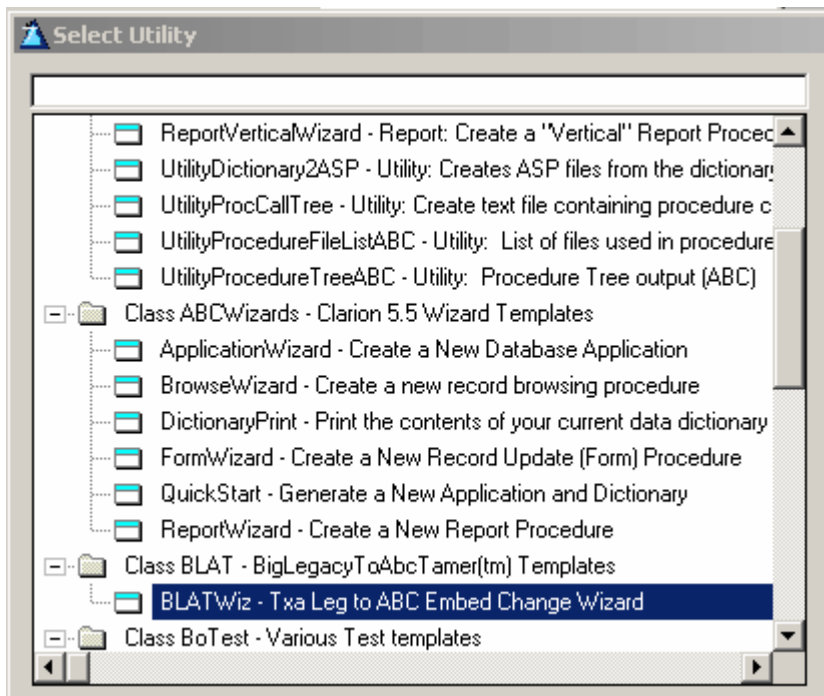


4. Run BLAT against the "TXA"

Select from the menu
Application/Template Utility/

or use the hot key combination **Ctrl+U**





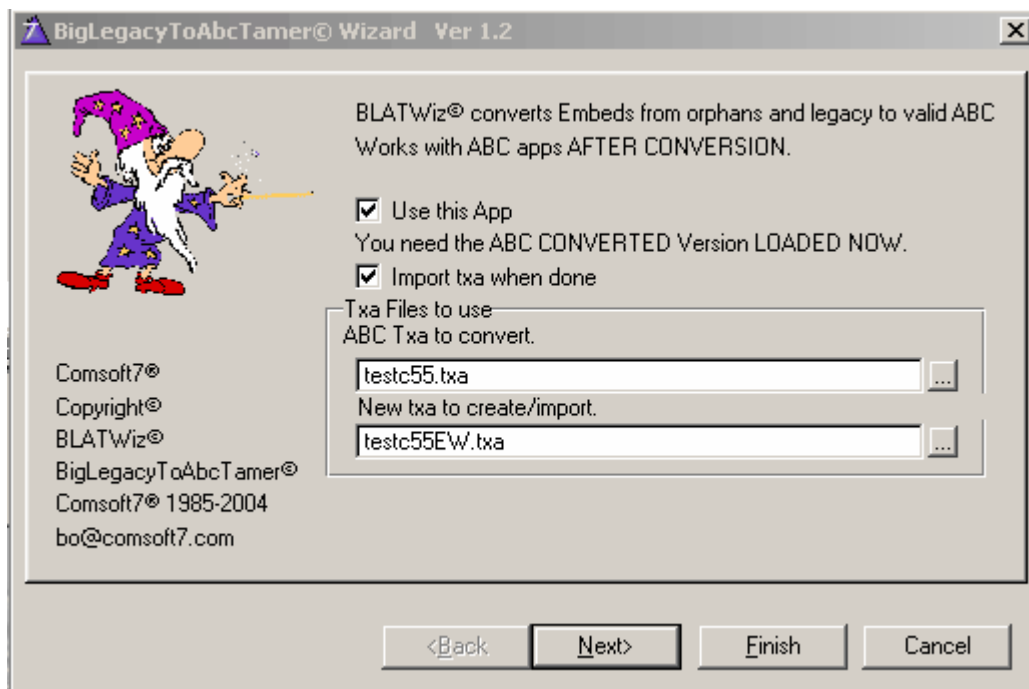
You can choose to automatically import the new TXA or do it manually after BLAT Runs.

Click on "Use this App" and "Import txa when done" to automatically import the converted TXA file.

You must have the correct abc app loaded to import.

You must select the exported txa file for BLAT to convert.

You can accept the default name for the new txa to be created, or select a new name.



Then select Finish, and wait a few seconds/minutes as BLAT does its magic.

1.3 What BLAT Does

BLAT is a utility template that runs against an exported TXA and moves specified Legacy and orphaned embeds into the new ABC embed Points.

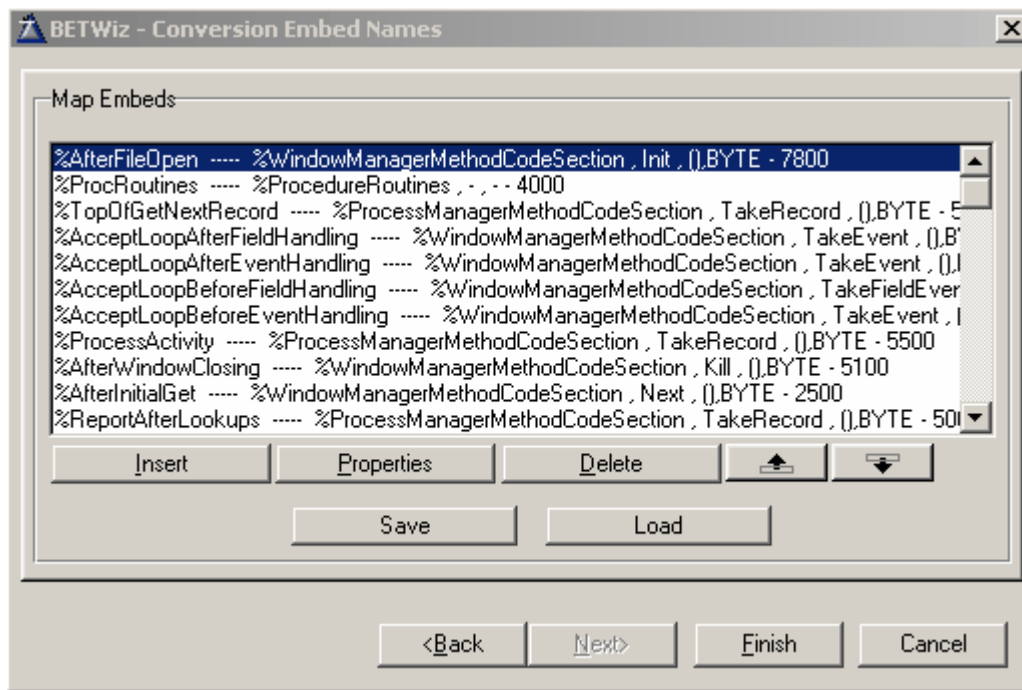
It can only move the Legacy and Orphaned embeds that have new ABC embeds defined. You will still have to decide what to do with the Legacy and Orphaned embed points that do not have an ABC equivalent.

BLAT has 77 embed points predefined that will be moved to the new ABC equivalents.

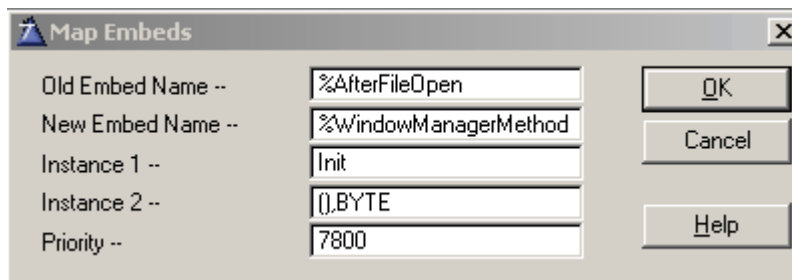
Usually there is no need to modify them, but you can salt to taste.

These 77 predefined embed points can be modified on the "second page" of the wizard utility template, or with the [BLAT.exe](#) provided with the template.

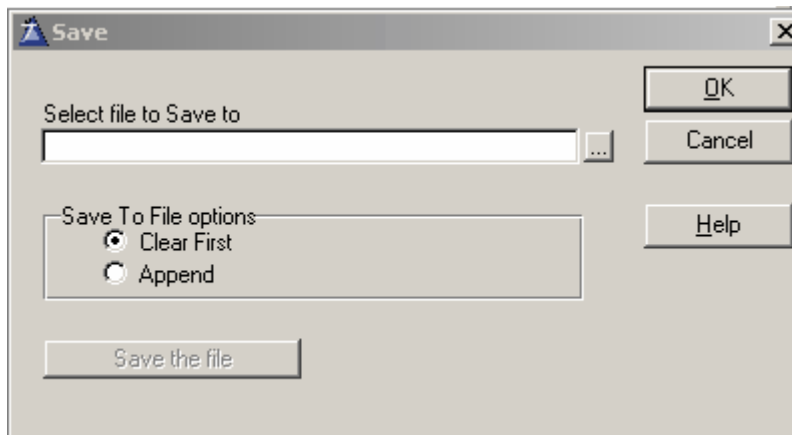
If you modify them, you have to save them, and load them each time you use the utility.



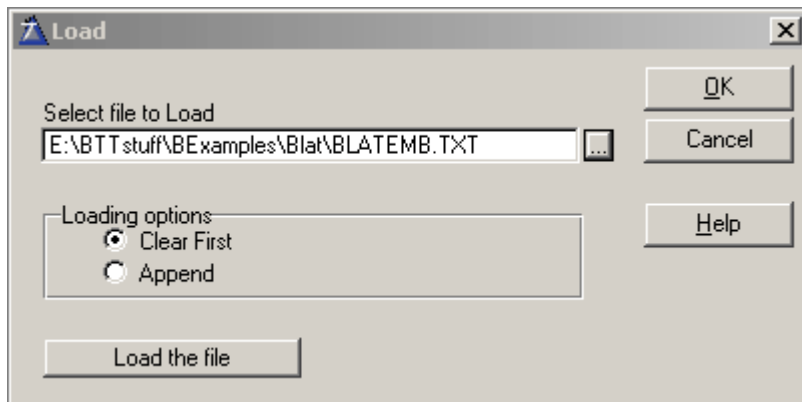
Properties or insert Prompts



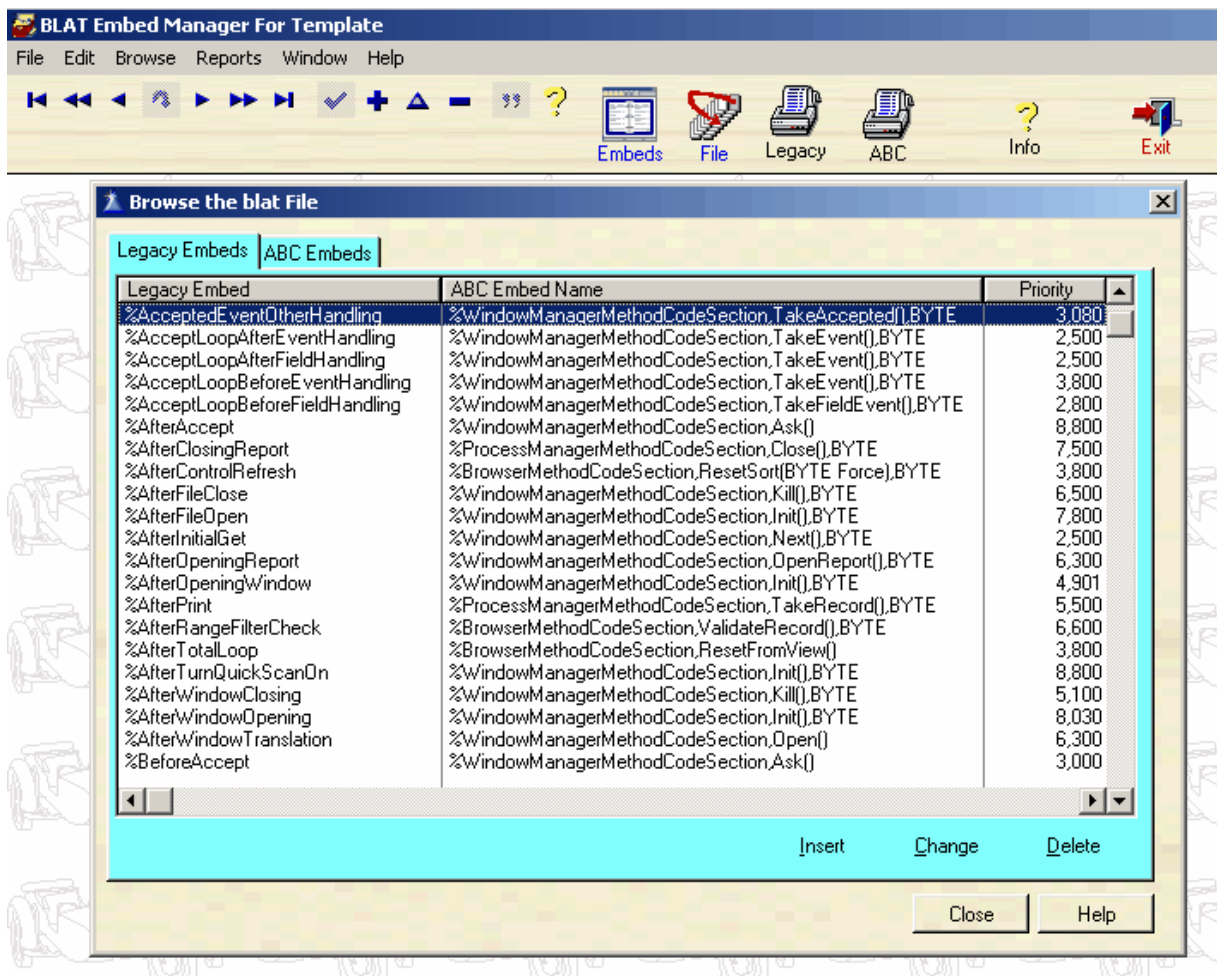
Save prompts



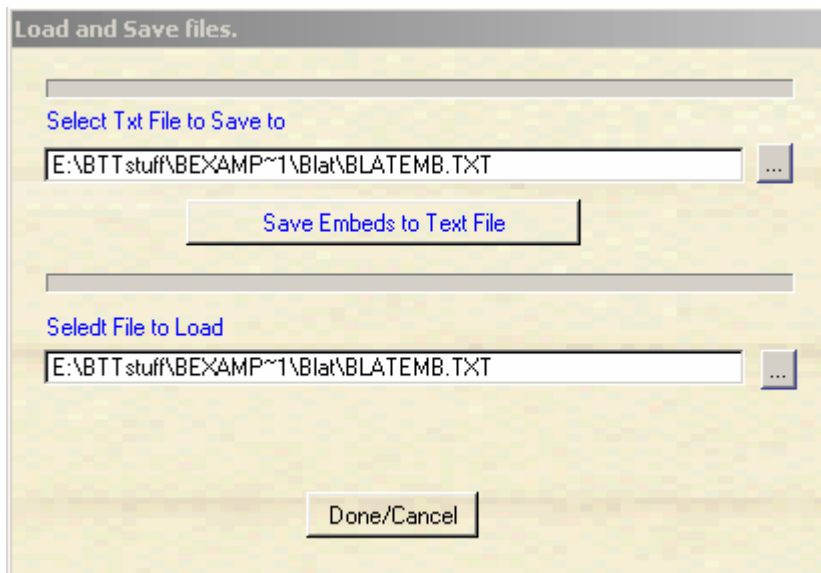
Load Prompts



1.3.1 Blat.exe



Embeds button on the toolbar brings up the embeds as shown above.



File button allows importing and exporting text files to/from the tps database to interact with the template.

2 Report print options are provided sorting by the Legacy embeds or ABC embeds that are moved.

1.4 What Is BLAT

What is BLATWiz© Template .

BLATWiz(tm) is a Legacy(Clarion) to Abc Template Embed Converter Wizard Utility.

BLATWiz(tm) moves Legacy(Clarion) embedded code to the new correct valid ABC embeds. BLATWiz(tm) does not change any code in the embeds, but just moves them to the new embeds. BLATWiz(tm) can do weeks of manual labor in minutes.

BLATWiz(tm) currently moves Legacy(Clarion) embeds into the following embed sections.

- %WindowManagerMethodCodeSection
- %ProcedureRoutines
- %ProcessManagerMethodCodeSection

BLATWiz(tm) moves Legacy(Clarion) events and control events into the following embed sections.

- %WindowEventHandling
- %ControlEventHandling

- %ControlHandling
- %BrowserMethodCodeSection

BLATWiz(tm) does its magic on the txa file.

BLATWiz(tm) runs from within the IDE.

The app must first be converted from the Clarion Template Chain to ABC via the Softvelocity Convertor.

BLATWiz(tm) will use the selected external txa file. It creates a new txa file (%Application & 'EW.txa') with the changes, and can optionally import the results to the current loaded app. You can change the name of the destination txa file.

It leaves the external txa file as is, and creates a new txa file with the changes. You have control over the name of the new txa file.

Growth of the txa files have been experienced up to 12%, and is to be expected because a legacy (Clarion) embed of

```
EMBED %BeforePrintPreview
[DEFINITION]
[SOURCE]
PROPERTY:BEGIN
PRIORITY 4000
PROPERTY:END
====BLAH,BLAH CODE
[END]
```

converts to an ABC embed

```
EMBED %WindowManagerMethodCodeSection
[INSTANCES]
WHEN 'AskPreview'
[INSTANCES]
WHEN '()'
[DEFINITION]
[SOURCE]
PROPERTY:BEGIN
PRIORITY 1300
PROPERTY:END
====BLAH,BLAH CODE
[END]
```

[END]
[END]

It is self documenting and creates a log file BLATEW.TXT showing the new embeds created and where they came from by procedure. It literally moves over a thousand embeds on larger apps.

You can examine BLATEW.TXT to see where code was moved to, and use it to fine tune the embeds by spinning the priorities to move the code up or down the embed tree if necessary. The embed destination points were selected from the source code section of the embeditor to place them immediately above or below the Legacy(Clarion) embed point in the app.

Run time for the conversion depends on the complexity and size of the txa being converted, and resources on your development machine. Small apps run in minutes, large apps (4meg txa) can take up to an hour on slow machines. A message box shows the progress of the conversion so you know it didn't hang.

An embed manager app is included, Blat.exe that allows changes to the embed points that are preloaded in BLATWiz(tm). You can change them, add to the list, save to a text file, and load them into BLATWiz(tm) at runtime.

NOTE: Care must be taken when adding embed conversion points as using the wrong syntax or embed point can cause errors in the txa. Usual problems encountered are errors loading the txa, or an extra [END] in the [EMBED] section causing the window to disappear in the affected procedure. You can only add embed points that are listed in the seven embed group destinations above.

For the braver sports, and a more permanent solution, you can change and/or add embed conversions in the template itself in

```
#!-----  
#GROUP(%BldConv)  
#!-----
```

The format is as follows, %BoEmbeds is a unique number identifying the embed conversion point. %BoOldEmbed and %BoNewEmbed should be self-explanatory. %BoInst1 and %BoInst2 are the instances of the new embed point to be substituted or added to the old embed point. %BoPriority was 4000 for all Legacy(Clarion) embeds and gets converted

to the one entered here. A minus '-' is used where instances are not needed so the template knows it is not needed. For Control event handling, the control or Browse:instance is loaded in addition at runtime. Orphaned and embeds where there are no valid replacements are ignored. They have to be dealt with manually on an individual basis. You can only add destination embed points that appear in the seven embed groups listed above.

```
#ADD(%BoEmbeds,1)
#SET(%BoOldEmbed,'%AfterFileOpen')
#SET(%BoNewEmbed,'%WindowManagerMethodCodeSection')
#SET(%BoInst1,'Init')
#SET(%BoInst2,'(),BYTE')
#SET(%BoPriority,7500)
```

```
#ADD(%BoEmbeds,61)
#SET(%BoOldEmbed,'%ControlPreEventCaseHandling')
#SET(%BoNewEmbed,'%ControlHandling')
#SET(%BoInst1,'-')
#SET(%BoInst2,'-')
#SET(%BoPriority,500)
```

```
#ADD(%BoEmbeds,72)
#SET(%BoOldEmbed,'%BrowseBoxAfterUpdate')
#SET(%BoNewEmbed,'%BrowserMethodCodeSection')
#SET(%BoInst1,'ResetFromAsk')
#SET(%BoInst2,'(*BYTE Request,*BYTE Response)')
#SET(%BoPriority,2000)
```

How it works:

BLATWiz(tm) loops through the txa looking for embeds. It creates a temp file Pass1.\$\$\$ with an intermediate txa containing markers for the embed points. It creates a marker for each of the seven embed groups contained above for the first occurrence found in each procedure. It then adds all other replacement and original embeds to the right occurrence saving all the embedded code in tempfiles.
('Blat' & %BlatFileCounter & '.\$\$\$')

Then BLATWiz(tm) loops thru the temp file replacing the markers with the new consolidated embed groups writing to the destination txa. Because of limitations in the the template language it must loop thru the tempfile once for each marker encountered. This could involve up to 7 loops per procedure in the worst case scenario. I.E. 700

loops thru the temp file on a 100 procedure app.

1.5 Contact Us

Contact Bo

bo@comsoft7.com

Website

<http://www.comsoft7.com>

Purchase

<http://www.developerplus.com/publishers/tools/ksa/bigtamer/index.asp>