

FutureBasic 7.1.1 marks a change in the minimum macOS to run it. Please note the change below.

FutureBasic 7.1.1 **requires macOS 12.0 or later**. Applications built with this version **will also require macOS 12.0+ to run**.

Translator

- New - Breakpoint panel buttons shortcuts:
'Continue program execution' = ⌘→
'Step over' = ⌘↑
- Fix - Use of the em-dash character in a BeginCXxx block would throw a compiler error. Multi-byte characters (like em-dash that starts with 0xE2, 0x80 just like the bullet character) in comments caused the character scanner to fail because it thought it was processing a bullet character.
- Change - Line and block comments within a BeginCXxx block are now *always* copied to the generated C code
- Fix - A bad `goto` label would sometimes crash FB

Editor

- Build Settings - The minimum 'Min deployment' allowed is now 12.0
- New - Project contextual menu item: *Exclude from Build*. Choosing this item will exclude the selected file from the build
- Fix - When the `next` statement was used on the same line as `for`, and the optional *IndexVar* specified after the next statement, the next line would be indented, e.g.

```
for x = 0 to 5 : v(x) = 1 : next x
    //.. indent
```

CocoaUI

- Renamed Functions. The old names for these convenience functions caused confusion, so they now have new names

OLD	NEW
StatusItemButtonSetImage	-> StatusItemSetButtonImage
StatusItemButtonSetTitle	-> StatusItemSetButtonTitle
StatusItemButtonSetToolTip	-> StatusItemSetButtonToolTip
StatusItemButtonSetAppearsDisabled	-> StatusItemSetButtonAppearsDisabled
StatusItemButtonSetActionCallback	-> StatusItemSetButtonActionCallback

Note: If your app uses any of these functions, please rename them ASAP because the old functions will be removed in the next FB release

Headers

- MTL4 headers added to the Metal framework
- New framework - ModelIO. This framework is optional and can be added to your app by: include "Tlbx ModelIO.incl"
- New framework - MetalFX. This framework is optional and can be added to your app by: include "Tlbx MetalFX.incl"
Note: Including this header also includes Tlbx Metal.incl
- Removed old unused Open GL headers (Tlbx glu.incl, Tlbx glut.incl, Tlbx gl.incl) as these are long obsolete in favor of other methods like Metal framework.

FutureBasic Reference

- Addition - `FBVersion` keyword

FutureBasic 7.1 marks a change in the minimum macOS to run it. Please note the change below.

FutureBasic 7.1 requires macOS 11.0 or later. Applications built with this version **will also require macOS 11.0+ to run.**

Translator

- New keyword/function - `FBVersion` returns the current FutureBasic app version as a 5 or 6-digit integer. For example, if the current version is 7.1, `FBVersion` returns 70100.

```
#if ( FBVersion >= 70100 )  
    // ...  
#endif
```
- Fix - A colon at the start of a line in a `BeginCXxxx` block was stripped
- Breakpoint window changes:
 1. Text can now be drag-selected and copied to the clipboard by cmd-C or clicking the  icon. If no text is selected, *all* the text is copied
 2. cmd-Q shortcut added to quit button
- New - The Xcode version is now displayed in the Build Log
- New - shorthand syntax for `peek` and `poke int` utilizing the bullet character '•' (option + 8 on the keyboard) now available

```
pointer p          // create pointer var  
int v, x           // create two int vars  
v = 57             // assign 57 to v  
  
p = @v             // put v's address in p  
x = •p             // shorthand for peek int( p )  
•p, 26             // shorthand for poke int p, 26 – put 26 the the address pointed to by p
```
- The `width` keyword no longer does anything so has been removed. Note: apps containing this keyword will now throw an error, so please remove it

Editor

- Fix - Double-clicking a snippet in the Library panel, when no open documents, would create a new file which included default document text
- Fix - A document's warning icon would intermittently fail to display
- Fix (hopefully) - 'undo' bug
- Change - Copying or Cutting a document's text no longer strips line indents from the copied text. Note: this change is related to the above 'undo' bug fix

CocoaUI

- New - `Metadata.incl` header

Headers

- New - *Security* framework/headers. This framework is optional and can be included with: `include "Tl bx Security.incl"`
- New - *Accelerate* framework/headers. This framework is optional and can be included with: `include "Tl bx Accelerate.incl"`
- New - *AppleArchive* framework/headers. This framework is optional and can be included with: `include "Tl bx AppleArchive.incl"`
- New - *Compression* framework/header. This framework is optional and can be included with: `include "Tl bx Compression.incl"`

FutureBasic Reference

- Updates

FB Examples

- Bug fixes and updates

20260325 FB 7.0.40

Translator

- Fix - **val** function failed to convert a binary constant (such as `0b0100011`) to decimal
- Fix - Double-clicking an error in Errors window would open the wrong source file and show wrong location.
- Fix - `PrintToWindowStatement()` creates code generating "incompatible pointer" for FBer builds.
- Internals Updates: Gradual move from Core Foundation MRC (manual reference counting) to NS ARC(automatic reference counting) continues.

Editor

- Fix - A project window's contextual menu Delete item is now disabled when no item is selected

FutureBasic Reference

- Appendix B - Variables - Updated pointer section and included example of using an asterisk for pointer dereferencing.
- **val** page includes a Hex, Octal and Binary usage examples with recommended syntax.
- Sharing nib-based examples on the list explained on the "Interface elements" page

20260223 FB 7.0.39

Translator

- Fix - If the source folder path to a *.metal file contained space characters, the *.metal file would fail to compile
- Fix - Double clicking warning/error in error window highlights wrong line in FB source window.
- New - FB version and build number now included in the build log.
- Update - voucher file creation changed to include FB's build number. Internal only.
- Fix - Linker warnings/errors restored to Errors/Warnings window and not just the Build Log
- Fix - 'cp' (copy) of an app's help files failed with a warning due to unescaped paths.
- Update - FBer's help folder now copied with a Cocoa command instead of the previous 'cp' command. Internal only.
- Fix - Make Xcode Project failed to copy Info.plist from built .app when "output file" deployed in FB source.
 - Internals Updates: Gradual move from Core Foundation MRC (manual reference counting) to NS ARC(automatic reference counting) continues.

Editor

- Fix - Choosing Command > Help Book > Create... would put a copy of the Help Book in the project's Resources group. This is not necessary as the Help Book is automatically included via the Info.plist entry
- Change - Choosing Command > Help Book > Create... no longer asks for the Help Book name and Help Book directory name. These names now default to the project name
- New - QuickLook now displays the contents of FutureBasic documents, i.e. click an FB document in the Finder, and hit spacebar. Works on project window files too!

CocoaUI

- The 'Cannot find a view with tag...' alert now shows the name of the function where the bad view tag was used
- Fix - The MTKView **_mtkViewDrawableSizeWillChange** dialog event failed to fire

Headers

- Updates and additions

FutureBasic Reference

- New - Appendix K - Build System - overview of Translator, Compiler and Linker Error/Warning Message

FB Examples

- FB Examples/HelpManager/HelpDemo2 using NSHelpManager was broken. Updated to reflect current standards.
- Bug fixes

20260109 FB 7.0.38

Translator

- Fix - Returning a record from a function using the `return` statement would error.
- Fix - unmatched double quotes in a `print` statement (e.g. `print @"";"@""`) produced an infinite translation loop, causing FB to hang and possibly crash.
Please see list post on 4-January-2026 "*FB hangs on malformed strings*" for more information.

- Fix - some error messages missing or only reported in build log.

- Fix - `include` statement for an optional framework, such as the one below, failed with errors.

```
include "Tlbx Accessibility.incl"  
AccessibilityCustomActionRef ref
```

- Fix - file i/o statements such as `print#`, `write file` failed.

- Fix - `textfinder.incl` not recogized.

- Internals Updates: Gradual move from Core Foundation MRC (`manual reference counting`) to NS ARC(`automatic reference counting`) continues.

These changes are hopefully invisible during FB's use but Brian's coding mistakes might surface.

- Fix - double clicking an error/warning in error window would not highlight in source. (late addition to FB 7.0.37 release) - Brian)

- N.B. clang's (via FB's) `analyze` sometimes fails to identify ***all*** leaks (i.e. "Potential leak of an object....." messages) in a source file.

It is not an FB issue as it is reproducible in both FB and Xcode.

CocoaUI

- Update - Previously, calling a function with a tag value of a view that did not exist would fail silently. This now throws an alert

Headers

FutureBasic Reference

FB Examples

20251115 FB 7.0.37

Translator

- New - a constant name can now contain one or more underscore (`_`) characters

Note: This feature will break existing apps that add constants by:

```
value = _const1_const2
```

- New - `xref` alternate syntax added - See FutureBasic Reference for details

- Fix - Using `mda_swap` more than once in the same local function would throw an error

- Internals Updates: Gradual move from Core Foundation MRC (`manual reference counting`) to NS ARC(`automatic reference counting`).

- Fix - double clicking an error/warning in error window would not highlight in source. (late addition after initial release - Brian)

Editor

- New - If an Xcode project folder exists in the FB app's build folder, Command > Clean Build Folder gives the option (via alert) to also delete the Xcode project folder.

CocoaUI

- Update - 'control does not exist' alert now displays the called function name where the error occurred
- Header updates

Headers

- New - Tlhx CFStringEncodingExt.incl. Note: This header is optional

FutureBasic Reference

xref page update

Other updates

FB Examples

Examples/Numbers/GMP_MPFR_BigInt_demos directory

20251013 FB 7.0.36

Translator

- Fix - Including a *.h, *.c, *.m, etc. file when either the source folder or filename contained spaces would throw an error
- New keywords - [trim](#), [trimL](#), [trimR](#) - Removes leading and/or trailing characters from a CFString. See FutureBasic Reference for details

Editor

- Fix - Pasting code containing nibtext into a document would sometimes fail
- Fix - A project's 'Create Prototypes File' contextual menu would fail to create a prototype for a function declaration containing line-continuation character(s).

CocoaUI

- Fix - Typos and missing delegate event in the ORSSerial framework. Note: Some of these changes may break existing apps
- New - [mda_array](#) statement. See FutureBasic Reference for details
- New - TableView convenience functions added: [TableViewResizeColumnAtIndex](#) and [TableViewResizeAllColumns](#)
- New - A window's print view 'Copy' contextual menu can now be disabled by calling [WindowPrintViewDisableMenu\(wndTag\)](#)
- Header updates and additions

FutureBasic Reference

- [popupbutton](#) page updated
- [trim](#), [trimL](#), [trimR](#) pages added
- [mda](#) page updated
- [NSLog](#) page now includes macro usage with an example

FB Examples

- Bug fixes and updates

Website

- The GMP / MPFR frameworks and headers are now available on the website's Downloads page

20250909 FB 7.0.35

Translator

- A single-line comment (//) can now appear after a line-continuation character (↵), e.g.

```
if a == 1 && ↵ // comment
  b == 2
  ...
end if
```

Note: A comment after a line-continuation character is not possible when it's within a literal string.

```
CFStringRef s = @"abc ↵ // not allowed
def"
```

- Copying resources error message now more informative. Provides directories involved too.
- New - [nearlyequal](#) and [nearlyzero](#) keywords (functions) for comparing double values. See FB Reference for details
- The [concat](#) function has been extended to handle CFURL, IndexPath, CFData, MapTable and HashTable. See FB Reference for details
- Fix - Including an alias to a private framework would fail to copy the framework to the app bundle
- Fix - The [str](#) and [str\\$](#) functions would return an incorrect value when processing negative fractions
- New - The leading space applied to a string returned from the [str](#) and [str\\$](#) functions can now be turned off (or on) by calling `STRSetLeadingSpace( bool )`
- Breakpoint window
The 'esc' key is equivalent to clicking the Exit button
Quit button (⌘) added. This terminates the app
- Fix - For loop inline declaration of an unsigned xxxx loop variable would throw an error, e.g.

```
for unsigned long i = 2 to 10
  print i
next
```
- Fix - Including *.h, *.c, *.m, etc. files when the source folder contained a space in its name would error
- Fix - [Write File](#)'s address version translated incorrectly, causing a build error.

Headers

- CTFontManager.incl added to CoreText headers
- ORSSerial framework for serial port calls is now available and headers are here: FutureBasic.app/Contents/Resources/Headers/CocoaUI/ORSSerial

A simple very early demo proves the framework is working:

```
include "ORSSerial.incl"

local fn GetUSBPorts
  ORSSerialPortManagerRef ref = fn ORSSerialPortManagerShared
  CFArrayRef ports = fn ORSSerialPortManagerAvailablePorts( ref )

  print @"Available Serial Ports:"
  for ObjectRef port in ports
    print port
  next
end fn

fn GetUSBPorts

HandleEvents
```

Editor

- Fix - When a function name, variable name, etc. contained the word 'next', and was on the same line as a [for](#) statement, subsequent lines would fail to indent

CocoaUI

- Fix - The backspace character (⌫) is no longer [printed](#) to a window
- Feature removal - The previous FB release let you set a control's height to 0 to auto-fit it to the macOS's recommended size. This has turned out to be buggy, so the feature has been removed. Set implicit heights instead to keep your UI looking right
- The `TextLabel.incl` header has been removed and its functions transferred to `TextField.incl`
- [TextLabelWithTag](#) and [TextLabelExists](#) functions are now deprecated. Please use [TextFieldWithTag](#) and [TextFieldExists](#) instead
- The `IKImageView` delegate is no longer installed by default. To receive user dialog events for the `IKImageView`, you must now call `IKImageViewSetDelegate` after the `IKImageView` has been created or loaded from a nib

- The UIImageView tag value is now sent to the user on dialog `_imageViewImagePathChanged` event 'tag' parameter

FutureBasic Reference

- [REM](#) page updated
- [str](#) page updated
- [nearlyequal](#) and [nearlyzero](#) pages added
- Colon separator page added to table of contents
- [Write File](#) page updated to clarify usage of its two flavors

FB Examples

- Bug fixes and updates

20250720 FB 7.0.34

Translator

- Fixed - The latest clang would throw a warning for C structs with the `__packed__` attribute
- Fixed - Prior release failed if a comment (starting with a double slash `//`) contained double quotes (`"`)
e.g. `Str255 t : t = "a "" b "" "" "" "" " // prints: a " b "" comment containing double quotes failed.`
- Deprecation - Older `proc` keyword duplicates `@fn` syntax. Now flagged as deprecated in this release and will be removed in the next.
Please switch from `proc "MyFunction"` to `@fn MyFunction`
- Update - The compiler option `-Wmissing-braces` is now turned off by default. It silences clang warnings for things like:
`MKMapRect r = {1,2,3,4} // r = {{1,2},{3,4}} nested braces now optional`
 Note: The option can be overridden by adding this line to your app: `#build CompilerOptions @"-Wmissing-braces"`
- New - CoreML *.mlmodel files are now compiled and placed in the built app's Resources directory. Include them by `include resources "MyModel.mlmodel"`
- Update - The translator now finds private frameworks in the User Libraries folder
- New - Two dispatch block keywords added: `dispatchapply` and `dispatchasync`. See FutureBasic Reference for details

Headers

- Update - The CGImageSource and CGImageProperties APIs are no longer included as standard. If your app uses them, you must now include the ImageIO header: `Tlbx ImageIO.incl`

Editor

- Fixed - Jump to definition would sometimes open the wrong header file
- Fixed - The state of the 'Auto-open last project' checkbox in Settings/General was not included in an exported file
- New - A previously exported settings file can now be imported by any of the following:
 - The Setting window 'Import' button
 - Dragging the file onto:
 - the FB Settings window
 - FB's idler window
 - FutureBasic.app in the Finder
 - FutureBasic.app in the Dock
 - Opening the file via File > Open...

 Note: When any of the above are used, the Settings window will be displayed and refreshed

- Removed - The 'Clean Folder' feature was causing undue anxiety for some users so has been removed
- Fixed - Adding a *.framework folder to the User Libraries folder would crash FutureBasic
- New - Autosave documents option (see Settings/General). With this option tuned on, documents will be saved automatically at intervals determined by the system.

💡 Note: This feature opens up the option to rename a file via its titlebar.

CocoaUI

- Update - To help locate where a 'Class error' alert is triggered, the name of the called function is now displayed in the alert
- New - CoreML framework. This framework is optional and included by: `include "Tlbx CoreML.incl"`
- Fixed - The *pdfDoc* parameter of the `pdfview` statement was ignored if the view did *not* already exist
- Update - The `pdfview` statement's *pdfDoc* parameter can now be any of the following:
 - pdfDocRef
 - url of a pdf file
 - path to a pdf file
 - name or path to a pdf file in the app's Resources directory
 - pdf data (CFDataRef)
- Fixed - The *index* parameter of the `radiogroup` statement was ignored
- Update - Calling `ImageWithSystemSymbolName` now sets the image name to the given symbol name
- Fixed - Calling `nibwindow` a second time on a window which has its *releaseWhenClosed* flag set would fail to reload the window
- Fixed - A small size (16 x 16), spinning style `progressindicator` would fail to display in its window
- Update - The `pathcontrol` statement *url* parameter can now optionally be a *path*
- Update - The image parameter of the `SegmentedControlSetSegment` function can now be any of the following:
 - The name or path to an image resource
 - An ImageRef
 - A system symbol name
 - The url of an image file
- New keyword - `segmentedcontrolsegment`. This can be used instead of the `SegmentedControlSetSegment` function. See FutureBasic Reference for details
- Fixed - `SecureTextFieldEchoBullets` would always return `NO`
- New keywords for views introduced in macOS 26.0 - `backgroundextensionview`, `glasseffectview`, `glasseffectcontainerview`. See FutureBasic Reference for details
- Automatic control sizing.
 - In preparation for control size changes in the next macOS release, many control statements (e.g. button, checkbox, textfield, etc.) now support automatic sizing.
 - If you pass 0 for the height (and in some cases the width), FutureBasic will automatically calculate the optimum size at runtime based on control type and macOS version.

Example:

```
button 1,,, @"My Button", (13,13,100,0)
```

Creates a button with height of 32 on macOS 15.0 and earlier, and 24 on macOS 26.0

FutureBasic Reference

- Update - `pdfview` statement
- Update - Notes added to *rect* parameter for all controls affected by size changes in macOS 26.0
- Other updates and additions

FB Examples

- Bug fixes and updates

20250406 FB 7.0.33

Translator

- Deprecated: per the on-list announcement the following calls now generate warnings. Please discontinue using them.
 - `defsng letter1 [-thruLtr1] [,letter2 [-thruLtr2]...]`
 - `defdbl letter1 [-thruLtr1] [,letter2 [-thruLtr2]...]`
 - `defstr letter1 [-thruLtr1] [,letter2 [-thruLtr2]...]`
 - `defint letter1 [-thruLtr1] [,letter2 [-thruLtr2]...]`


```
deflong letter1 [-thruLtr1] [,letter2 [-thruLtr2]...]
```

Please note: defstr, in addition to the letter1, letter2 syntax, supports a 2nd syntax for defining printed string lengths when using bin/bin\$, hex/hex\$. This 2nd syntax is NOT deprecated and there will be no warnings for its use. This second syntax is described more in the FB References.

- Fixed: Translator would incorrectly reduce spaces between words to one in a C literal string.
e.g. `char cs(30) = 'T was fair'` would change the string content to: `'T was fair'`
- Fixed: Translator would incorrectly recognize an apostrophe as end of a pascal string.
e.g. `Str255 s : s = "Twas 'but"`
- Fixed: Escaped double quotes (`\`) would not translate correctly.
e.g. `char cs(30) = 'T \"was\" fair'` would fail with a build error.
- Fixed: Single line comment (e.g. `//`) on same line as literal string caused error.
e.g. `Str255 t : t = "a "" b "" "" "" "" " // prints: a " b "" comment containing double quotes WAS failing.`
- Fixed: Maximum line length increased to 2000 single byte characters.
translator treats continued lines (`↵`) as the same line,
making it easy to blowout the maximum characters array when declaring either a C or pascal string literal.
- Fixed: `fn FBXOpenSerialPort` within `OSX SerialIO.incl` was rewritten to eliminate poor coding practices and make it more readable.
- Fixed: Incorrect generated C code for NAND/NOR keywords caused incorrect results
- Fixed: Cast warning for `FBIniteAEEvents`
- Fixed - The `defstr` keyword would not highlight, and incorrectly issued a 'deprecated' warning
- Fixed - The latest clang release (17.0.0) would throw a warning when the user on dialog function did not specify ALL of its params (`ev`, `tag`, `wnd`, `obj`).
Similar for the `on appevent` and `on menu` functions
- The `ucs` and `ucc` functions now handle 32-bit values. See FutureBasic Reference for more info
- The `ucs` keyword syntax has been expanded to allow multiple, comma-separated code points. See FutureBasic Reference for more details
- The `len` keyword syntax has been expanded to support an optional *option* parameter. See FutureBasic Reference for more details
- The `kill` and `rename` keyword syntax has been expanded to support optional *result* and *err* parameters. See FutureBasic Reference for more details

```
[BOOL result = ] kill url [, @err ]  
[BOOL result = ] rename oldURL{ to | , } newURL[, @err ]
```

Headers

- Updates and additions

Editor

- Fixed - Choosing *File > Open* would sometimes fail to display an open dialog
- Fixed - A document's contextual menu would occasionally contain two 'Paste' items
- Choosing *Copy with Attributes* when there's no selected text in a document now copies the entire document's text to the clipboard
- New - Optional document toolbar item -  Share
Clicking this toolbar item copies the selected text (with attributes) into a new Mail message. When there's no selection, the entire document's attributed text is copied to the Mail message
- Fixed - A project's *Create Prototypes File* option would fail to recognise functions that did not have all lowercase `fn` and/or `local` keywords. It would also miss function declarations that had the `clear` keyword

- Fixed - A document's toolbar *fn* menu would fail to display functions that did not have all lowercase **fn** and/or **local** keywords
- New - Menu item: *Find > Find Selection in Project*
- New - Document contextual menu item: *Find Selection in Project*
- Change - Only links that are in comments are displayed in a document's contextual menu
- New - Errors window now has a contextual menu Copy item. This puts the clicked row text onto the pasteboard

CocoaUI

- Fixed - All the functions in the *Text.incl* header did not verify the widget class, and would sometimes crash when calling an unsupported function
- New - When intercepting an **openpanel** or **savepanel** accessory view **_btnClick** event in the user on dialog function, the *wnd* param now contains the **openpanel** or **savepanel** tag
- New String Unicode utility functions. See String.incl header for more details
StringUnicodeLength
StringUnicodeWithCodePoint
StringUnicodeWithCodePoints, etc.
- New - The **input** function/statement now has a **_formatPassword** option. This can be used alone or combined with one of the other format constants. See FutureBasic Reference for more details

FutureBasic Reference

- **kill** and **rename** updated
- **len** updated
- **ucc** and **ucs** updated
- **if** re-written
- **nand** and **nor** updated
- **available** updated
- **breakpoint** updated to remind of *Debug > Deactivate Breakpoints* menu
- other updates

FB Examples

- Additions:
 Dates and Times/DateUtils/ (long threatened by Brian)
 OpenPanel AccessoryView
- Dispatch group demos updated
- Other additions and bug fixes

20250222 FB 7.0.32

Translator

- Build log now displays the MacOS and build settings: Architecture, Min deployment, Base SDK
- Use of **index\$** arrays now throws up deprecated warning. They will be removed soon, so please switch to CFArray or mda
- Fixed - Mutable array shorthand setter (**gMyArray[1] = @"bravo"**) would throw an error with a global array
- Fixed - The following would throw errors:
CGRect r = **CGRectZero**
CGSize s = **CGSizeZero**
CGPoint p = **CGPointZero**
- The FB translator now compiles CoreData *.xcdatamodeld files. Include them with: **include resources "Model.xcdatamodeld"**
- The **globals "somefile"** statement is now deprecated and will be removed soon. Please use **include** instead
- New syntax for the undocumented **mda_find** function. See FutureBasic Reference for more info:
long index = **mda_find** [tag,] object [, range]
- Fixed - the **write file** statement would fail with error when the @err param was used

- Entering an index value of -1 in the `mda` array function now returns the last object in the array (-2 returns the penultimate object, etc), e.g.

```
mda(0) = {"a","b","c"}
print mda(-1) // prints 'c'
```

- Fixed - Using the old `page` keyword as a variable name would throw an error
- Fixed - Using `mda_count` without a tag value in a function call would throw an error, e.g.

```
fn CFRangeMake(0,mda_count)
```
- Fixed - Issuing an alert statement with button titles, for an alert that already exists, would duplicate the buttons with the same titles
- Fixed - short-hand CFMutableArray setter failed with translation error on global arrays but not local.
- Deprecated - per the on list announcement the following calls now generate warnings. Please discontinue using them.

```
defsgn letter1 [-thruLtr1] [,letter2 [-thruLtr2]...]
defdbl letter1 [-thruLtr1] [,letter2 [-thruLtr2]...]
defstr letter1 [-thruLtr1] [,letter2 [-thruLtr2]...]
defint letter1 [-thruLtr1] [,letter2 [-thruLtr2]...]
deflong letter1 [-thruLtr1] [,letter2 [-thruLtr2]...]
```
- Enhancement - `stop` statement can now include a filename as a CFString (not pascal). See FB Reference.

Headers

- CoreData, CoreMedia and Metal headers updated
- Photos framework added
- Fixes(leaks, warnings and sloppy code) to OSX SerialIO.incl. Mostly for fn FBXSerialGetCountAndNames

Editor

- Errors window - The errors table can now be scrolled horizontally
- Fixed - Hitting the return key on a line that starts with an open paren '(' would hang FB
- Fixed - Document 'save as' dialog extension popup did not update the filename field
- Document 'save as' dialog now shows file extension by default

CocoaUI

- These long-deprecated types have now been removed:
`CocoaEventRef`, `CocoaMenuRef`, `CocoaPasteboardRef`, `CocoaWindowRef`, `CocoaControlRef`
- Document.incl and DocumentController.incl headers are no longer included as standard. In the unlikely event that your app calls any of the functions in these headers, you must now add this line to your app or include the header in your project

```
include "Tlbr Document.incl"
```
- Fixed - MenuSelectGroupItem would fail to correctly set a menu item under a section header
- Fixed - `StringIsLike` and `StringIsCaseInsensitiveLike` gave the wrong result

FutureBasic Reference

- Updates - some are extensive. 'if' was almost completely re-written. A little more clarity for available.html

20241222 FB 7.0.31

Translator

- Fixed - CGRect parenthesised assignment was broken in the last build, e.g. `CGRect r = (10,20,30,40)`
- Deprecated(for 4 years) `long if` and `xelse` keywords removed. If your apps still use these, you will need to change to `if` and `else`
- New - `if` now has an 'else if' option
- A function's return type can now optionally precede `local fn` or `def fn`, e.g.

```
double def fn MyFunction
double local fn MyFunction
```

- The parameters for **Syntax 1** of the `line input` function/statement now has an optional *encoding* parameter. This can be one of the `NSStringEncoding` constants listed in CocoaUI's String.incl header (default = `NSUTF8StringEncoding`.)

```
cfstring = line input fileID [, encoding [, @err ]]
```

Note: This change may break existing apps that call this function using the `@err` parameter. The fix is simple: Insert an extra comma between `fileID` and `@err`.

- FB's `rnd` function can now take an optional 'min' parameter, e.g.

```
randomInteger = rnd( [min,] max ) // return random integer in the range min to max inclusive
```
- New - C String support enhancement:
 - `appendCStr` keyword. Non-destructive append.
 FutureBasic Reference explains more on the new keyword and C strings usage.

Headers

- Updates and additions
- OSX SerialIO.incl updated to use CFArray of CFStrings instead of an array of pascal strings. Its functions perform the same logic. Also see Brian's FB list post on 18-Dec-2024.

Changes that will impact your code are:

1. The standard global arrays of pascal strings are now global CFMutableArrays of CFStrings. Array names are: `gFBSerialName`, `gFBSerialInName`, `gFBSerialOutName`. Any code referencing them will need updating to process CFStrings from a CFMutableArray.
2. The corresponding `close` for the open "C" will deallocate those CFMutableArrays of CFStrings. This will need consideration if the program has multiple concurrent open "C" ports open. In that case leaving the first open until both ports can be closed would be one approach.
3. Code calling:

```
fn FBXOpenSerialPort( path as CFStringRef, speed as long, parity as long, stopBits as long, charSize as long, origOptions as ^termios ),
```

Please note the first parameter "`path`" is now a CFString, not a pascal string.

Editor

- Fixed - Build log progress indicator failed to display on the first build of an app after FutureBasic relaunch
- Fixed - Find in Project window would intermittently select the wrong item when clicked
- Fixed - Choosing Grayscale Slider or CMYK Sliders in the color assets color panel would fail to change the color values
- Change - minimum deployment supports only macOS 10.13 and later. Announced on list 10-Oct-2024.

CocoaUI

- `MutableDataReplaceBytesInRange` parameters have changed. The *length* parameter has been removed.
Note: This change will break existing apps that call this function. Please use the replacement function: `MutableDataReplaceBytesWithLength` instead.
- Fixed - The inkey panel would not move with its parent window when `%(x,y)` parameter used.
- Fixed - `ControlFormatterSetIsPartialStringValidCallback` was broken.
- `ControlSetFormat` function and `input` statement *caseOption* parameter name has changed to *option*. This parameter now has the following extra constants:

```
_formatInteger           // numbers and '-' prefix
_formatUnsignedInteger   // numbers
_formatFloat             // numbers, '-' prefix, and one decimal point
_formatUnsignedFloat     // numbers and one decimal point
```

Note: When one of the above is specified, the *characters* and *pass* parameters are ignored.

- Fixed - `ProgressBarIsDisplayedWhenStopped` and `ProgressBarSetDisplayedWhenStopped` would throw an error.
- Fixed - `WindowPrintViewSetAutoscrolls` would not work properly when the input panel was in use

FutureBasic Reference

- Updates
 - `appendCStr` page added
 - * Appendix D - Numeric Expressions updated to included casts and ULL modifiers for integer literals.
 - Open page updated to correct typo for use of "method" (changed to pascal string from CFString)
 - `rnd` updated to explain new parameter usage and implications of not using the new parameter.

FB Examples

- New example: Graphics/Metal/Metal Demo
- Updates

20241004 FB 7.0.30

Translator

- New - `ucc` (unicode character code) keyword. This is the `CFString` -> `unichar` equivalent of the pascal `asc()` function. See FutureBasic Reference for more info
- New - `ucs` (unicode character string) keyword. This is the `unichar` -> `CFString` equivalent of the pascal `chr$()` function. See FutureBasic Reference for more info
- The `include`, `include library` and `include resources` statements can now optionally use a CFString literal. Note: pascal strings are still supported
- `mda_remove_nulls` is deprecated. Use `mda_compact` instead
- New - `concat` keyword. Although initially designed to concatenate CFStrings, it supports other types, including CFArray, CFDictionary, CFSet, etc. See FutureBasic Reference for more info
- New - C String support enhanced:
 - Single line C string declaration and initialization added: `char myStr(6) = 'abcdef'`
 - `concatCStr` keyword. Concatenates two or more C strings. FutureBasic Reference explains more on the new keyword and C strings usage.
- New - CFMutableArray and CFMutableDictionary 'setters', e.g.


```
array[2] = @"Delta"
dict[@"key2"] = @"Bravo"
```
- Fixed - `read#` and `write#` failed to write 4 and 8 byte integers (NSInteger, long and int) correctly
- Fixed - `bit` function now handles 32 and 64-bit integers
- Fixed - Using the CFString `len()` function with a pascal string or container would crash the user's app. This now throws an error at translation time
- New - Multiple inline record assignments now supported, e.g.


```
CGPoint pt1 = {20,690}, pt2 = {20,20}, pt3 = {50,150}
```
- Change - Omitting the inner braces in an inline `CGRect` assignment now generates a harmless compiler warning

```
CGRect r = {10,20,30,40} // compiler warning
```

Two options to silence the warning:

(1) Use parentheses instead of braces

```
CGRect r = (10,20,30,40)
```

or use nested braces

(2) Include the inner curly braces

```
CGRect r = {{10,20},{30,40}}
```

- New - Inline C-type array assignment support, e.g.


```
long a(5) = {1,2,3,4,5,6}
```
- New - pre/post inc/dec operators have now come to FB. See FutureBasic Reference, inc/dec for more info


```
long num = 0
long a(2) = {123,456,789}
print a(num++) // 123
print a(num)   // 456
```


- Fixed - The `spc` keyword would throw an error when used in a print (to window) statement
- New - Fast-enumeration of `mda` array, e.g.

```
mda(0) = {"Alpha","Bravo","Charlie"}
CFStringRef string
for string in mda
    print string
next
```
- New - For loop variable declaration. With this release, we can declare a loop variable on the same line, e.g.

```
for short i = 1 to 10
    print i
next
```

Notes:

1. When declared this way, the loop variable is *not* available outside the loop
2. There cannot be another variable with the same name as the loop variable in the same local function

- New - `breakpoint` statement. The breakpoint statement is used to pause execution of the running program. This allows inspection of current values of specified variables. See FutureBasic Reference for more info
- New - Shorthand assignments

```
a &= b    // a = a & b
a |= b    // a = a | b
a >>= b   // a = a >> b
a <<= b   // a = a << b
a %= b    // a = a % b
a ^^= b   // a = a ^^ b
```

Headers

- Additions and fixes

Editor

- New - Project window reopening on FB relaunch can now be turned on/off in Settings/General
- Line spacing option added. This can be set in the Text Editing tab of Settings, or the Editor/Line Spacing menu, or the optional 'Line spacing' document toolbar item
- Fixed - Setting the line number text colour for dark mode was broken
- New - Document contextual menu item 'Autoresizing Helper...'. Animates the autoresize effect of the chosen springs and struts on a subview. Also allows pasting of the autoresize constants into a document
- Fixed - Navigate > Jump to Definition menu item was broken
- Recents window updated
 1. Contextual menu added: *Hides On Deactivate*, *Floating Panel* and *Size* items
 2. Displays file icons
- Submenu added to document contextual menu copy item

```
Copy > Copy
      Copy with Attributes
```
- Declarations inside a `begin globals` / `end globals` block are now indented
- New - Project find option to ignore quoted text
- New - Create assets colorset. See 'View > Assets > Color'
- New - Project files list window. This window shows a sorted list of all files in a project. The window is displayed, via 'Files List' item in the project's contextual menu, or via the optional toolbar item. The project files list window has a contextual menu with 'Hides on Deactivate', 'Floating Panel', and 'Size' items.

CocoaUI

- The `SoundNamed` function now accepts a path to a sound resource located in a subfolder in the app's resources directory, and the sound object's name is automatically set to the filename.

- The **SoundPlay** function now accepts a sound resource name or path
- Change - A new **panel** built with its origin set to {0,0} is now centred on the screen to match **window** behaviour
- Fixed - In a nib, a contextual menu assigned to a window's content view would be lost
- Widget statements with an image param, and many XxxSetImage functions, now accept any of the following:
 - An ImageRef
 - An image resource name
 - A path to an image resource
 - A system Image name
 - A system symbol name
- Fixed - An **alert**'s buttons would sometimes fail to appear in the panel
- Fixed - Changing the attributes of an existing **levelindicator**, using the **levelindicator** statement, would sometimes shift its location in the window
- Fixed - Calling the **alert** function in an **if** statement, e.g. `if ( alert(1,,, @"OK;Cancel") == NSAlertFirstButtonReturn )` would throw a translation error

FutureBasic Reference

- Appendix B- Variables updated with examples of one-line declaration/assignment of variables
- **bit** function reference updated to clarify overall use and clarify how floating point input is used
- **concat** page added
- **concatCStr** page added
- **for** page updated with loop variable syntax and examples
- **mid** function/statement updates
- **ucc** and **ucc** pages added
- Other updates

FB Examples

- Several new String demos (Examples/Strings and Text/String) for **concat**, **mid** insert, **ucc**, **ucs** and **concatCStr**

Other

- QuickLookUI.framework is now weak-linked into FB itself (but not built apps) to prevent crashes in older macOS versions. ask on list for any questions.

20240724 FB 7.0.29

Translator

- The **sound** keyword has been resurrected to work with functions in the Sound.incl header. See FutureBasic Reference for more info
- Variable type-identifier suffixes (TIS) are not supported (by default) in this version of FB. However, those who are still using the outdated variable TIS in their apps can turn them back on by overriding a new translator constant:


```
override _allowVariableTIS = _true
```

Also, to help transition to modern code, an additional constant override is provided to warn of TIS usage and location:

```
override _warnOfVariableTIS = _true
```

Note: overriding **_allowVariableTIS** may interfere with other FB commands that use TIS symbols (% , & , ! , #)
- As of this release, subroutines (gosub/return) are turned off by default. However, those who like to wallow in nostalgia can turn them back on by overriding a translator constant:


```
override _allowSubroutines = _true
```
- The **return** keyword can now be used to exit a function. See FutureBasic Reference for more info

Note: this option is not available if subroutines are turned on (see above)
- The translator now issues an error if a nib file, named in a nibwindow, etc. statement, is not in the user source folder. This helps to avoid a user app crash
- A build will now report an error if a **container** is sent to **mid\$** (**mid\$** is for pascal strings only). **mid\$** fails when a **container** contains more than 256 characters.

Headers

- All Sound.incl functions now accept either a tag value or SoundRef
- * Fixed - Subs DefUsr.incl change usage of TIS (#) vars to explicit 'double' type
- Other updates and additions

Editor

- FB now warns of multiple copies of the FutureBasic app found on the same computer. This warning can be turned off in Settings/General 'Warn of multiple FB apps'
Note: this option is only available for computers running macOS 12.0 or later
- Fixed - Build Settings/Compiler options change without first hitting the return key would result in the edited string being lost
- Fixed - Build Settings 'Set all to defaults' was broken
- Fixed - Find in Headers window - The 'Word' checkbox did nothing. It has now been replaced by a popup button with options: Contains, Whole word, Starts with, Ends with.

CocoaUI

- Fixed - The *menuID* and *itemID* parameters in [MenuItemSectionHeaderWithTitle](#) function were unused so have been removed. Note: this change will break existing apps that call this function
- [UserUnixTaskExecute](#) function name changed to [UserUnixTaskExecuteWithArguments](#) Note: this change will break existing apps that call this function
- Fixed - When a windows content view is set to flipped, [ViewsIsFlipped](#) function would return NO
- Extra on dialog constants added for window mouse events: [_windowMouseDownDragged](#), [_windowMouseDownMoved](#), etc.

NSLog

- An indeterminate circular progress indicator can now be turned on and off in the NSLog window titlebar by calling [NSLogProgressStart](#) and [NSLogProgressStop](#)

FutureBasic Reference

- [lprint](#) and other old, unsupported print-related keywords removed
- [sound](#) page updated

FB Examples

- New Sound/Sound Statement
- Bugs fixed

20240630 FB 7.0.28

Translator

- Fixed - The question mark character (?) in a BeginCXxxx block would sometimes throw an error
- Fixed - The [pen](#) statement with *dash pattern* value would hang FB.
- Fixed - Other [pen](#) statement bugs
- Fixed - Rebuilding a Universal app after code changes would sometimes fail to include the changes
- The [catch](#) statement parenthesised exception parameter is now optional
- The [stop](#) statement now optionally accepts a CFString literal message. e.g. [stop @ "Help!"](#)
- Alternative syntaxes are now available for the [open](#), [close](#), [line input](#), [read file](#) and [write file](#) statements. See FutureBasic Reference for more info


```
[ result = ] open @"i|o|a", fileID, url [ , @err ]
[ result = ] close [ fileID [, @err] ]
[ cfString = ] line input fileID [ , @err ]
cfString = read file fileID [ , @err ]
[ result = ] write file fileID , cfString [ , @err ]
```
- Shorthand syntax has been extended to include CGPoint, CGSize, CGRect and CFRange, e.g.

```
CGPoint pt = {10,20}
ValueRef ptVal = @(pt) // translates to: ValueRef ptVal = fn ValueWithPoint( pt )
```

- Print to window now allows printing of CGPoint, CGSize, CGRect and CFRange, e.g.

```
print @(point)
print @(size)
print @(rect)
print @(range)
```

The above will be printed to the window in this format:

```
NSPoint: {10, 20}
NSSize: {30, 40}
NSRect: {{10, 20}, {30, 40}}
NSRange: {0, 100}
```

Headers

- Added subclass names (in comments) for nib widgets
- The eight standard color constants, `_zYellow` to `_zWhite` have now been added to the CocoaUI.incl header and are no longer hard-coded (hidden from view) in the translator

Editor

- View menu - Show Recents item added
Choose this menu item to display a floating panel listing the contents of the File/Open Recent menu
Click an item to open and/or bring the document to the front
- Document toolbar - Recents item added to optional toolbar items. Clicking this is same as choosing View/Show Recents
- Project toolbar - Recents item added to optional toolbar items. Clicking this is same as choosing View/Show Recents
- Create Prototypes File' added to projects contextual menu. This item inserts a 'Prototypes' group below Globals and generates a 'Prototypes.prot' file containing prototypes (def fn Xxx) of all the project's local functions
- Fixed - Trashing a file in a project, when its window was open, would leave the file's window open
- The user 'Help Menu Items' directory can now contain subdirectories (one level deep). Subdirectories appear as submenus in FB's Help menu
- Fixed - Choosing a bookmark in a project's function/bookmark popup (option-click) would fail to find the bookmark

CocoaUI

- `mkmapview` - The subclass name "MapViewSC" is deprecated. Use "MKMapViewSC" instead when subclassing in a nib
- `wkwebview` - The subclass name "WebViewSC" is deprecated. Use "WKWebViewSC" instead when subclassing in a nib
- Fixed - `WindowPrintViewScrollToBottom` was intermittently broken
- Window print view can now be set to autoscroll printed text by calling `WindowPrintViewSetAutoscrolls( wndTag, YES )`
- Fixed - The `scnview` statement with `scene` parameter would fail with error
- `mda_append` - This new statement adds the contents of a CFArray to an mda array. See FB Reference for more info
- Header updates and additions
- Fixed - The `input` and `inkey` floating panels would not move with their parent window
- Fixed - When the `input` or `inkey` floating panels were dismissed, their parent window was not made key
- Fixed - `MutableDataNew` function was broken
- With this release, there is no need to build an `openpanel` or `savepanel` invisibly before calling ancillary 'setter' functions, e.g.
`// openpanel -101 // not necessary`
`OpenPanelSetCanChooseFiles( 101, NO )`
`url = openpanel 101`

- With this release, there is no need to build an `alert` panel invisibly before calling ancillary 'setter' functions, e.g.

```
// alert -201 // not necessary
AlertSetIconNamed( 201, @"MyIcon" )
alert 201
```

- Fixed - `AlertSuppressionButtonState` would sometimes return the wrong value

- XxxProperty functions renamed.

```
XxxSetProperty          -> XxxPropertySet
XxxRemoveProperty       -> XxxPropertyRemove
XxxRemoveAllProperties    -> XxxPropertyRemoveAll
```

Note: The old functions are still usable but are flagged deprecated so please transition to their replacements as soon as possible

- Convenience getter and setter functions added to ObjectProperty, AppProperty, MenuProperty, MenuItemProperty, WindowProperty, ViewProperty

```
XxxPropertyBool, Long, Double, Rect, Point, Size, Range, BookmarkURL, Bytes
```

- Fixed - Adding a segment to a segmented control would sometimes throw an 'unknown window' error

- Hide/show tooltips functions added

<code>AppSetToolTipsHidden</code>	hides/shows tooltips in all menu bar menus, windows, toolbars, views and subviews
<code>MenuSetToolTipsHidden</code>	hides/shows tooltips in all menu items in a menu
<code>WindowSetToolTipsHidden</code>	hides/shows tooltips in all views and subviews in a window
<code>ToolbarSetToolTipsHidden</code>	hides/shows tooltips in all toolbar items in a toolbar
<code>ViewSetToolTipsHidden</code>	hides/shows tooltips in a view and all its subviews

Note: Whilst the above works for tableview/outline view column headers and row views, it cannot toggle tooltips in outline items that are hidden in a collapsed parent.

FutureBasic Reference

- Added subclass names for nib widgets
- Other updates

20240407 FB 7.0.27

Translator

- Comparing pascal strings with `<`, `>`, `==`, etc. would return `-1` when true. This now returns `1` (`_true`).
- The maybe keyword would return `-1` for true. It now returns `1` (`_true`)
- Resolved - Using a non-constant in the system function would throw an error, e.g. `system(7)`
- Fixed: Literal CFString containing copyright symbol (©) failed when used in a #plist compiler directive
- Two late changes made it into a late build of FB 7.0.26
 - Fixed: Compiler warning with `allowedClassesForRestorableStateKeyPath:`
 - Fixed: Sheet window failures because FB wasn't handling multiple windows in one nib file correctly.
- Line Input# would fail to notice EOF when a file's text content does not include a CR/LFs. Issue was masked by an incorrect use of EOF with the NOT bitwise operator.
- New: Logical 'not' symbol(exclamation '!') introduced: `while ( !eof(1) )`
Bitwise 'not' is unchanged to avoid breaking user code.
- New: percent symbol (%) is a synonym for the FB 'mod' operator:
`a % b` is the same as `a mod b`
- Fixed: inline Boolean assignment generated errors


```
long c = 11 & 2 // error!
long c = a & b, d = a & b // error!
```
- Fixed: no space between variables and ampersand character caused an error


```
c = a&b // error!
```

Editor

- New document contextual menu item - *Make Default New File Text*. Choose this menu item to make the current document text the default when creating a new document (⌘N)
- Project Find/Replace panel - minor tweaks to UI

CocoaUI

- The `pen` statement now has an optional *shadow* parameter. See FutureBasic Reference for more info
- Resolved - `print @(col, row)` would fail with error
- `_windowMain` added to the `window` function's selectors
- CocoaUI alerts can now be suppressed and redirected to NSLog and/or Console.app, by calling
`CUI_SetSuppressAlerts( YES )`
N.B.: Suppressing CocoaUI alerts is not recommended for the release version of an app
- Resolved - many UI statements would throw an error when a Boolean param was an expression. For example, the *enable* param in the button statement: `button 1, value == 12` // error
- `openpanel` and `savepanel` statements *fileTypes* parameter can now be a UTType or an array of UTTypes. See FutureBasic Reference for more info
- The following statement params can now take either a `CFStringRef` or a `CFAttributedStringRef`:
`button`, `checkbox`. `radiobutton` *title* param
`textlabel` *text* param
`menu` *title* param - note: only applies to a menu item, not the menu itself
- Resolved - MKMapItem function name typo fixed. Was `MKMapItemSetPhoneNumbewr` now:
`MKMapItemSetPhoneNumber`

FutureBasic Reference

- Appendix G - Symbol Table
Dim x;4. marked as obsolete/unsupported
@[for creating a CFArray added
@{ for creating a CFDictionary added
- Appendix R - rounding options - Errors corrected and example from math.incl added

Examples

- Views and Controls/MKMapView/Coordinates from address
- New demo to archive FB Records at: FB Examples/Files/Archiving FB Records

20240210 FB 7.0.26

Translator

- `autoreleasepool` / `autoreleasepoolend` keywords added. Provides same support as their Objective-C progenitors without having to code C
- Resolved - `timerbegin` was generating an unnecessary semi-colon on entry to the ^block. Luckily clang ignored it but we can't count on that behavior
- lipo crashed when the translator used the wrong app name internally. The external symptom is the built app won't launch.
- DynamicClear function added to Dynamic array support. Without deallocating the dynamic array's memory, it clears the array to zeros from index zero through the highest index accessed by user code.
- Resolved - setting the `pen` color with a `ColorRef` contained in an mda array would require: `mda_object(3)` with this release, it can be simply: `pen 1, mda(3)`
- The fn `ROUND` function from the Subs DefUsr.incl header was simply a wrapper for the `int` function so has been removed. Use the `int` function instead.
- Removed ampersand's (&) use/support as a function name's TIS. User code fix: change `local fn MyFunction&` to `local fn MyFunction`
note: '!', '%' and '#' are untouched.
- Resolved - Using the bitwise '&' operator with a function's returned value would throw an error
`value = (fn MyFunction & 12)` // error

Headers

- New optional include, fenv.incl header, supplies constants to control rounding for many of the rounding-related functions within match.incl

Editor

- Snippets panel - there's now no need to first select a row to enable contextual menu items
- Resolved - Clean Build Folder would sometimes fail to delete the previously built app

CocoaUI

- Resolved - Using a zero tag value in the `window` statement would create an invisible, inaccessible window
- Resolved - Calling the window function with a *positive* selector value of 6 or higher would return the app's key window. With this release, calling the window function with a selector value that's out of range (i.e. greater than 6) throws up an alert.
- Resolved - `StringEnumerateSubstringsInRange` options parameter was incorrectly assigned a CFStringRef type (should be NSUInteger). This made it impossible to stack options.

- With this release, calling nibwindow with the same nib name as used in a previous nibwindow statement now removes the previous nib from CocoaUI's internal nib list.
- The `rect` statement now has an optional *cornerRadius* parameter
- The `rect` or `oval` statements, *with no parameters*, now reset all graphics attributes to default values

NSLog

- `NSLogHide`, `NSLogShow` functions added. `NSLogHide` hides the log window but it continues to receive messages. The log window remains hidden until `NSLogShow` is called or the app is relaunched.

FutureBasic Reference

- HandleEvents updated to show its correct placement when long-running (background or main thread) tasks are involved
- New Appendix R - Rounding Options
- Updates to Fix, Frac and Int to clarify the operation and mention alternative calls in math.incl (back to automatically included)
- Dynamic updated to clarify operation and improve its readability
- DynamicClear page added
- Ternary Operator page added
- User Interface (UI) element placement/position is explained on the Interface elements help page
- Other updates

Examples

- Bug fixes

20231229 FB 7.0.25

Translator

- `run` command updated to accept a path or url returned from a function, e.g.

```
run fn WorkspaceURLForApplication( @"Safari" )
```
- Resolved - The `str` and `str$` functions would strip any zeros that appeared immediately after a decimal point. e.g.

```
print str(1.001)
```

would print 1.1
- Resolved - Use of the `str` function would create a leak
- `Open "Unix"` support removed. Use the newer `unix` function.
- Numeric Literal Changes per section 1.1 of FB Reference appendix C.
 - Binary Integer literal: `'&X'` deprecated. Use the new `'0b'` syntax instead. Translator issues a deprecation warning.
 - Hexadecimal Integer: `&H`, `&` are deprecated. Use the `'0x'` format instead. Translator issues deprecation warnings.
 - Octal Integer: deprecated with a future planned removal. Translator issues a deprecation warning.
 - See FB Reference appendix C (some in appendix G) for details.
- `_maxLong` constant reinstated. Warning: It will be defined as `NSIntegerMax` which is a 19 digit integer (9223372036854775807).
If you see FB Help suggesting this constant's use for an FB dynamic array, use a literal constant of one (i.e. `'1'`) instead.
- Dynamic arrays are updated to allow more indices. Number of indices now governed by an eight byte integer instead of the prior 4 byte integer. All other aspects of Dynamic arrays (data types it supports, supporting functions, general operation etc.) remains the same.
- Single apostrophe use for a single-line comment now emits a deprecation warning. Replace with double slash (`//`).
- A build no longer includes the Carbon framework by default. If you need the Carbon headers and need to link to the Carbon framework, use the `'include library "Carbon"'` statement
- Hundreds of bit comparisons replaced with logical comparisons
- `_gFBQuit` eliminated. Do not rely on this global to quit an app. It's gone.
- * `_zTrue` constant eliminated. **Note: this change will break existing apps using this constant**
FBers can either a) change `_zTrue` to `_true` if the underlying logic supports it or declare a `_zTrue` and set it to minus one.

Editor

- Possible fix for intermittent crash when closing a document window
- Cut, Copy, Paste items added to document contextual menu
- Resolved - cmd-double-clicking a user constant in a document would fail to jump to its declaration

CocoaUI

- A window's `'print'`ed text can now be copied to the clipboard via contextual menu item
- Function name changes - `ProgressIndicatorIsDisplayWhenStopped` and `ProgressIndicatorSetDisplayWhenStopped` have been changed to `ProgressIndicatorIsDisplayedWhenStopped` and `ProgressIndicatorSetDisplayedWhenStopped` to match Obj-C method names
Note: this change will break existing apps that call these functions
- The alert character sequence (`\a`) can now be used with the `print` statement. When this sequence appears within a string, a modal alert is displayed

- The alert character sequence (\a) can now be used with the `printf` statement. When this sequence appears within the *format string*, a modal alert is displayed
- Resolved - WKWebViewUIDelegate was broken
- Header updates and additions

NSLog

- The alert sequence '\a' can now appear anywhere in the format string
- alert '\a' and backspace '\b' character sequences are now stripped from the output, thereby removing gremlins
- Resolved - The backspace '\b' and tab '\t' sequences are now recognised when they appear next to the alert '\a' sequence
- Resolved - `NSLogAlert`, and its equivalent '\a', would display the format specifier characters ('%ld', '%@', etc.) instead of their values

FutureBasic Reference

- NSLog help page added
- A coding examples added to a few UI widget pages
 - Changes to appendix C and appendix G related to new deprecations (see Translator notes above) and clarifying the '#' symbol's operation
- Write Dynamic help page updated with runnable FB code

Examples

- Bugs squished
- New examples:
 - Task Management/ProgressObserver
 - Processes and Threads/OperationQueue/OperationDependency
 - Processes and Threads/Thread/SimultaneousThreads
 - Menus/MenuItemSlider/

20230927 FB 7.0.24

Translator

- New optional header - `sysctl.incl` contains convenience functions for getting system information such as CPU type
- The *color* parameters of the `oval`, `pen`, `rect` and `text` statements can now be either a `ColorRef` or one of FB's built-in constant names: `_zRed`, `_zGreen`, `_zBlue`, etc.
- Resolved - the `open "O"` statement would fail to truncate an existing file

Editor

- Resolved - Project Find/Replace panel 'swap' button would sometimes fail to display on macOS 10.15 or earlier
- 'Ignore Comments' checkbox added to Project Find/Replace panel
- Resolved - The `else` keyword would sometimes not indent correctly
- Document windows
 - Standalone document window frames:
 - a. use a new storage method.
 - b. won't use previously stored frame coordinates.

The 'Find' toolbar item was causing intermittent issues, so has been removed

- Internal changes to remove single quote (') use for bookmarks and divider lines
- Fixed: `cln` would fail to translate some literal strings like: `cln f = 0b00000011;`
- Fixed: `print` (to window) failed with various container-related calls such as: `print left$$`, `print right$$`, `print mid$$`, `print ucase$$`, `print string$$` and `print containerView`
- Fixed: Multiple successive calls (as in a loop) to `unix` would fail with a crash
- Resolved - An open parenthesis immediately following some keywords would mess up indenting. e.g. `if ( 1 )`

CocoaUI

- Resolved - The `textinsertionindicator` statement would not modify an existing indicator but would create an *extra* widget instead

FutureBasic Reference

- Updates

FB Website

- Broken links fixed plus other minor changes

20230816 FB 7.0.23

Translator

Fixed: Xcode projects created with FB's Make Xcode Project would fail with missing AppThings.h file
Fixed: info.plist was not being processed correctly resulting in a default plist being used instead of the intended one.
Fixed: Projects using path names using escaped characters (like spaces) would fail.
Fixed: Some keywords used to load an `mda` array failed occasionally.
Fixed: 'Make Xcode Project' process streamlined and extra help added when a framework (such as NSLog.framework) needs to be incorporated in the Xcode project.
New: new `unix` function (also statement syntax) replaces the older `open "unix"` statement. The advantages are:
1. When commands potentially produce a lot of output, the output size isn't limited to 255 characters.
2. No need to parse through the data line by line in order to collect it. It's all collected immediately.
3. No need to give it a file ID
4. No need to open or close the file ID
5. Facilitates nested calls when the output results are going to another function for additional processing.
6. Less code to write!
Those using the former Open "unix": it is deprecated and usage will warn of its pending removal.

Resolved: `printf !(col,row)` would throw an error

Resolved: The following keywords would sometimes fail to return the correct values: `bin`, `hex`, `lcase`, `space`, `str`, `ucase`

`mda` updates:

`mda_array` keyword added. This function returns the root array of the `mda` array. The returned value is a `CFMutableArrayRef`.

`mda_add` statement dim param is now optional for single-dimensional arrays, e.g.

```
mda_add(0) = 123
mda_add = 123 // same as above
```

`mda_count` dim param is now optional for single-dimensional arrays, e.g.

```
count = mda_count(0)
count = mda_count // same as above
```

`mda_remove` now has an optional *length* parameter, e.g.

```
mda_remove(2), 3 // removes 3 elements, starting at element 2
```

Negative `mda` tag values are no longer supported. Tags must now be in the range 0 through 2,147,483,647

Resolved - The following would either throw an error or return the wrong value

```
long intValue = mda(0)
double dblValue = mda(0)
```

Update: Window and widget statement 'bit mask' params can now have their values combined by using the vertical bar ('or' operator), e.g.

```
window 1, @"My Window", (0,0,550,400), NSWindowStyleMaskTitled | NSWindowStyleMaskClosable
```

Editor

New: Build Settings, 'Flatten nibs' option.

In previous versions of FB, 'flatten nibs' was always turned off, therefore included nibs could be viewed/edited in the built app.

With this option turned on, nibs in the built app cannot be viewed/edited.

Update: After a 'Clean Build Folder', the Build Log window (if visible) now remains open

CocoaUI

Resolved: An editable textfield in a nib window would not trigger the dialog `_btnClick` event

Resolved: When customising a toolbar, an item added to the toolbar would not trigger the `_toolbarItemClick` dialog event

Update: A negative tag value in the `nibview` statement now sets the view to hidden when the nib is loaded

Update: The `combobutton` statement has changed.

For convenience, a *menuItems* parameter has been added. See FutureBasic Reference for more info.

`combobutton` tag, enabled, title, image, menuID, *menuItems*, rect, style, wndTag

Note: This change will break existing apps that use this statement.

Examples

Bug fixes

New demos

FB Examples/Miscellaneous/Unix functionStatement/openUnix.fb

FutureBasic Reference

New: Appendix Q - "Make Xcode Project Description & Operation" added

New: Help for the new `unix` function/statement

Other updates and additions

Translator

- Resolved - bugs fixed in the [mda](#) experiment
- Resolved - Pascal string constants were broken in FB 7.0.21
- Resolved - [inkey](#) format bugs fixed
- Resolved - the 'container' variable type would highlight as a constant
- Resolved - An "analyze" followed by a "Build" or "Run" resulted in multiple build errors.
- Resolved - Type table size increased (from 2k to 8k) to accommodate optional includes. Some fixes to facilitate future size changes too.
- Ongoing - unix commands such as 'rm', 'touch', 'mkdir' converted to Cocoa code
- [Inkey](#) and [Input](#) keywords now have additional 'format' parameters. See FB Reference for more info
- The [menu](#) statement now has an extra *sectionHeader* parameter (requires macOS 14.0+). See Demo and Menu.incl header
- The [progressbar](#) statement now defaults to an *indeterminate spinning-style* indicator. To create a *bar-style* indicator, either call [ProgressIndicatorSetStyle\(tag, NSProgressIndicatorStyleBar \)](#) on the indicator or use the new [progressbar](#) statement. All functions in the ProgressIndicator.incl header work with both indicator styles.
- Ternary operator implemented. It mimics C's operator. Example: `c = ( a < b ) ? a : b`

Editor

- Resolved - If an [if...then](#) statement contained a backslash quote (`\`), the following line would be indented incorrectly.
- Resolved - Changing build settings for a standalone document would crash FB if a project window was open at the same time.
- Standalone FB files now remember their window positions and sizes
- 'Make Default Window Frame' item added to document contextual menu. Choosing this item stores the current window frame as default for new documents

CocoaUI

- Many CoreAnimation 'XxxSetPath' functions now accept either a [BezierPathRef](#) or [CGPathRef](#) as the *path* parameter
- Header updates
- Resolved - Calling WorkspaceURLForApplication would return a garbage value if the app did not exist
- XMLDocumenValidate function typo (note the missing 't') fixed. This correction will break existing apps that called this function.
- Many additions and new features introduced in macOS Sonoma 14.0, including...
 - menu
 - section header (see Demo, FutureBasic Reference and Menu.incl header)
 - selection mode (see Demo and Menu.incl header)
 - palettemenu (see Demo, FutureBasic Reference and Menu.incl header)
 - menu item badge (see Demo and MenuItemBadge.incl header)
 - popover
 - show relative to toolbar item (see Demo and Popover.incl header)
 - symbols
 - symbol effect (see Demo and SymbolEffect.incl header)

Examples

- Graphics/
SymbolEffect

- Menu/
MenuItemBadge
PaletteMenu
SectionHeader
SelectionMode
- Popover/
ShowRelativeToToolbarItem
- Strings and Text/XML/
XML Document
XML Parser2

FutureBasic Reference

- Added: two new help pages for Line continuation character and the colon character in the Symbols section.

20230523 FB 7.0.21

Translator

- Resolved - a variable named 'field' would throw an error
- Resolved - using backslash-double-quote (\ ") in an **if** statement would sometimes hang FutureBasic
- Resolved - including multiple *xcassets would result in the previous *xcassets being overwritten
- Resolved - using **exit fn** in an **if..then..else** statement would throw an error
- Resolved - 'print'ing a record, to a window, in its entirety would crash the app. This now throws a 'not supported' error
- Resolved - An inline CF type assignment would sometimes throw an error
- The equals sign ('=') in an '**exit fn** = *expr*' statement is no longer optional
- The **integer** var type is no longer supported. Please use **long**, **short** or **int** instead
- Removed: stringist and related keyword support.
- Removed: ThreadBegin/ThreadStatus keywords removed because supporting code was removed years ago.
- Changed: Min Deployment in the Xcode project after using "Make Xcode Project" now defaults to macOS 10.13 instead of 13.1
- Resolved: Several File Handle deprecations in FB's standard file I/O (such as Read File, Write File etc.) now choose older/newer methods based on execution macOS version running.
Also internal translator updates for same.
- Resolved: trailing line continuation character (↵) within a one-line comment (i.e. window 1 // something↵) caused translator to overlook next live source line.
- **inkey** and **input** have been resurrected (sort of!). See FutureBasic Reference and demos. Requires macOS 10.12+
- Experimental feature: **mda**
mda is an acronym for **multi-dimensional array** or **mutable dynamic array** - take your pick. It's all of these things and more. An **mda** array can be single-dimensional too. Is always global in scope, can hold NULL values and is identified by tag (rather than reference). **mda** arrays also support aggregate initialisation (aka shortcut population) as well. An **mda** stores all data as objects.

Simple example:

```
mda(0) = @"Alpha"
mda(1) = @"Bravo"
mda(2) = @"Charlie"

long i, count
count = mda_count
for i = 0 to count - 1
```

```
print mda(i)
next

HandleEvents
```

See demos and FutureBasic reference for more info.

Editor

- Resolved - re-indenting a document would give incorrect indenting when the cursor was next to, or within, some keywords
- File/File Info menu option now displays the number of characters and number of lines in a 'project'.

CocoaUI

- Header updates
- The user dialog event, `_viewResetCursorRects`, is now triggered by more control types. See demo

Examples

- New demos
 - Event Handling/
 - ResetCursorRects
 - Resources/Assets/
 - Color
 - Image
 - Strings and Text/
 - Cheap Line Numbers
 - Hyperlink
 - Inkey and Input/
 - inkey
 - input
 - Collections/Array/
 - mda
- Bug fixes

FutureBasic Reference

- Many updates including two new help pages for Line continuation character and the colon character in the Symbols section

.20230313 FB 7.0.20

Translator

- **TODO: This code snippet hangs. Removing the outer parens fixes it. Probably something to do with the backslash quote.**

```
if ( fn StringContainsString( @"" , @"\""" ) )
end if
```
- "Make Xcode Project" would fail because the source code path, and/or source file name, wasn't escaped. This was a ripple from other changes to move away from unix commands and use Cocoa.
- Fixed: writing (write# or read#) would fail on record fields. Thanks to TS for reporting
- Resolved - When a variable named *'timer'* was used as the return value from a `timerbegin` call, an **'incompatible pointer'** warning would be displayed when the variable was used inside the block.
- The following UI statements now double as functions. See each FB Reference page for more info.

Keyword	Return value	Return type	Equivalent function
button	state	NSInteger	ButtonState
checkbox	state	NSInteger	ButtonState
colorwell	color	ColorRef	ColorWellColor
combobox	text	CFStringRef	ControlStringValue
datepicker	date	CFDateRef	DatePickerDateValue
imageview	image	ImageRef	ImageViewImage
levelindicator	value	double	ControlDoubleValue
pathcontrol	url	CFURLRef	PathControlURL
popupbutton	selectedItem	NSInteger	PopUpButtonIndexOfSelectedItem
progressindicator	value	double	ProgressIndicatorDoubleValue

radiobutton	state	NSInteger	ButtonState
searchfield	text	CFStringRef	ControlStringValue
securetextfield	text	CFStringRef	ControlStringValue
segmentedcontrol	selectedSegment	NSInteger	SegmentedControlSelectedSegment
slider	value	double	ControlDoubleValue
stepper	value	double	ControlDoubleValue
tabview	selectedIndex	NSInteger	TabViewIndex
textfield	text	CFStringRef	ControlStringValue
textlabel	text	CFStringRef	ControlStringValue
textview	text	CFStringRef	TextString
tokenfield	text	CFStringRef	ControlStringValue

- Resolved - The [sourcelinenum](#) function would return the wrong value.

Editor

- **TODO - Double-clicking text and dragging cursor does not select additional words.**
- Resolved - Re-indenting a document would randomly indent text in comment blocks.
- New toolbar widgets - *Line Numbers* and *Re-Indent*. Note: These are not default items so must be added to the toolbar via View > Customize Toolbar...
- Syntax highlighting of [system](#) declarations are now displayed in same style and colour as constants.
- Resolved - Running a project that did not contain a build settings file (such as a legacy project) in its package would crash FB when built.
- Project Find/Replace All
 1. Replace All is now only allowed when affected files have been saved.
 2. Affected files that are open at the time of a Replace All are closed and reopened.
 Note: These changes only apply to **Replace All**. The **Replace** action is currently unchanged.
- Project Find/Replace window - Swap button added (between Find and Replace fields).

CocoaUI

- arrayWithContentsOfURL & arrayWriteToURL parameter changes.
 arrayWithContentsOfURL now takes an extra ErrorRef parameter

```
ArrayWithContentsOfURL( CFURLRef url, ErrorRef *err )
```


 arrayWriteToURL atomically parameter is now an ErrorRef.

```
ArrayWriteToURL( CFArrayRef ref, CFURLRef url, ErrorRef *err )
```


 Note: These changes will break existing apps. Please amend your code.
- Other header updates.

FB Help

- [instr](#) help page updated generally and for new search options.
- [sound](#) help page updated to provide general direction and example of using basic sound call.
- [openpanel](#) page updated with more information on `_openPanelDidEnd` event usage.

Examples

- Bug fixed in AVAudioPlayer Demo that spawns multiple timers without invalidating them.
- Bug fix - DistributedNotificationCenter.
- PlistSerialization & Emoji Fun fixed to work with updated arrayWriteToURL & arrayWithContentsOfURL functions.
- Two new dispatch_group demos - see FB Examples/Processes and Threads/Grand Central Dispatch

Website

- Tutorials - Working with nibs - All tutorials updated.

20230116 FB 7.0.19

Translator

- Unlike FB 7.0.18, second and subsequent builds (for a project or single source) compile only source changed since the previous build.
 - It honors the user's source files (i.e. doesn't recompile them if not changed), FB runtime (`_0_TranslatedRuntime.m`) and precompiled header (PCH)
 - This allows the user to turn around "runs" quickly when making changes
 - Works for single architecture builds (i.e. Intel --or-- Arm64) only - i.e. not universal
 - Facilitates quick turn around (runs) when making changes (e.g. in only one source file of several)
 - Universal builds always compile everything, so recommend selecting your native machine architecture (either Intel or Arm64/Apple Silicon) for the above capability
 - The second and subsequent build logs show only the files compiled and not the ones that didn't change and not needing a compile.

- New 4 byte integer types `int` and `unsigned int`
- 4 byte peeks/pokes available via `peek int` and `poke int`
 - 8 byte peeks/pokes handled via `peek long` and `poke long`
- `'xelse'` keyword now produces a deprecation warning. Please replace with `'else'`.
- FB's `instr` function now takes an optional `searchOptions` parameter. See FB Reference for more info.

Editor

- Resolved - Double-clicking an errors file (e.g. Errors.fberr) when FB is not running would crash.
- Resolved - FB would hang/crash when opening a file containing a period (full stop) in its name.
- Resolved - A document's functions drop down toolbar menu would list function declarations inside comment blocks.
- Resolved - A project's functions drop down menu (opt-click file) would list function declarations inside comment blocks.

CocoaUI

- Header updates

2021226 FB 7.0.18

Translator

- Print to window
 - Resolved - a window would sometimes have a slight (15px) horizontal scroll.
 - Resolved - Printing a `CGRect`, `CGPoint`, `CGSize` or `CGRect` field would fail.
 - Resolved - Text printed to a window would occasionally be clipped vertically.
 - Change - Printing a `CGRect`, `CGPoint`, `CGSize` or `CGRect` to a window is no longer possible.
- FB's translator now supports:
 - the `*=` (multiply equals) and `/=` (divide equals) operators. See *FutureBasic Reference/Appendix D - Numeric Expressions* for more info.
 - the `&&` (logical ANDing) and `||` (logical ORing) operators.
 - Those symbols(`&&` `||`) no longer support bitwise comparisons. See FB Reference's 'and' and 'or' keywords for more info on.

`exit fn` statement now takes an optional `'= expr'`. See FutureBasic Reference for more info.

Parallel Universal builds. Universal builds should be a bit faster than they were in the previous FB release.

- The Carbon headers (`#import <Carbon/Carbon.h>`) are no longer copied into the precompiled header. The Carbon.framework will still be linked for those who might need it but FBers who need those Carbon calls will have to write their own declares and FB toolboxes. Carbon headers absence should improve PCH build time a little.
- Superfluous file, `FBtoC_Prefix_gcc.h`, removed from build_goodies.
- `time` keyword fixed
- `'xelse'` keyword formally deprecated. Please update code to use the `'else'` keyword as `'xelse'` removal in the next release is likely.

Editor

- When saving a document or creating a project, the editor now warns if the save name contains illegal characters.
- Resolved - When creating a new project, a (harmless) NSLog window would be displayed.
- Resolved - A project window set to regular font size would display a gap below the toolbar.
- Resolved - Adding a file from the User Libraries folder to a project window would copy the file to the project folder.
- 'Show User Libraries Folder' item added to View menu.

CocoaUI

- Header updates

Preparation for the new logical 'OR is similar to the new logical 'AND' previously discussed on list

Recap of OR:
FB's current 'OR' is bitwise and has a double vertical bar (i.e. ||) synonym

For FB release 7.0.18 'OR' will still be bitwise but its synonym will change to the single vertical bar (i.e. |)

Logical 'OR' will be available via a double vertical bar (i.e. ||)

Preparation:

if your code currently uses a double vertical bar (i.e. ||) for its bitwise ORs, please change your code to use the OR keyword instead.

Example:

```
if ( a  || b )  then fn DoSomething  // uses '||' for bitwise OR

if ( a  or  b )  then fn DoSomething  // change to 'or' for bitwise OR
```

	Current	New	Action
Bitwise OR	OR	OR	Change ‘ ’ to ‘OR’ in bitwise ORs
Bitwise AND	AND &&	AND &	Change ‘&&’ to ‘AND’ in bitwise ORs
Logical OR	None		
Logical AND	None	&&	

20221114 FB 7.0.17

Translator

- An error is now displayed when the #build keyword is used in a project. Reminder: #build is only allowed in single-file apps.
- Resolved - The functions, bin, hex, oct and mki would throw an error when expression was boolean, e.g. bin(8 and 156).
- Resolved - Printing cvi to window would crash, e.g. print cvi(@"TYPE")
- Resolved - printing an integer or double value to a file with the print# keyword would throw an error.
- Resolved - iterative calls to instr\$ resulting in a start position greater than one could crash the app.
- Changed - poke long's shorthand character, the ampersand (&), is now the '~' character. **Note: This change will break existing apps using the ampersand.**
None of the other shorthand characters for peek and poke have changed and the shorthand deprecation warnings are removed.

NSLog

- [NSLogSetFrame](#) function added.

Editor

- The [Editor > Show/Hide Line Numbers](#) menu item removed. This option is now in the Preferences window Text Editing pane.
- [Code Snippet](#) items in the Editor menu are now a submenu.
- [Delete](#) item added to Code Snippet submenu.
- Library window - [Delete Snippet](#) item added to contextual menu.
- Resolved - Choosing an item in the optional Architecture toolbar popup would occasionally display the build settings window.
- Preferences window
 - 'Maximum recents' added to General pane.
 - 'Show line numbers' added to Text Editing pane.

CocoaUI

- **Changes that may break existing apps**
 - [DateStringWithFormat](#) function now takes a CFDateRef as the first parameter.
 - [SegmentedControlSelectedSegmentDoubleValue](#) function name has changed to [SegmentedControlDoubleValueForSelectedSegment](#). Also, this function now takes just the one parameter, *tag*.
 - [DistributedNotificationCenterAddObserver](#) - The *behavior* parameter of this function was unused so has been removed.
 - [ViewAdjustPageHeight](#) - The *oldLeft* parameter was unused in this function so has been removed.
- CoreImage headers are no longer included as standard. To use CoreImage functions in your app, add the header to the project or use: `include "Tlbx CoreImage.incl"`
- Resolved - Amending an existing box via the [box](#) statement would remove its title.
- Resolved - Amending an existing date picker with the [datepicker](#) statement would reset its elements to default (YearMonthDay).
- SceneKit framework added. See demos
- Additions to EventKit framework.
- Print to window now has vertical scrolling.
- Other updates and bug fixes.

New Demos

- Views and Controls/SCNView/
3D Moon Orbiting Earth
Shadowed Cube
Sun-Earth-Moon orbits
Tumbling Die with Pulse
Wireframe Sphere
- Dates and Times/EventKit/
EventKit Demo

20221003 FB 7.0.16

Translator

- Using the [block](#) keyword on a global variable now generates a *"block is only allowed on local variables"* error.
- Previously, each block variable or static variable would have to be declared on line of its own. With this release, they can be declared on the same line.

Previously:

```

block long a
block long b
block long c

static long sMyVar
```

```
static long sAnotherVar
```

Now:

```
block long a, b, c // a, b and c are block variables
```

```
static long sMyVar, sAnotherVar // sMyVar and sAnotherVar are static variables
```

- Resolved - Using the optional comma in the `#plist` statement would cause the value parameter to be ignored.
- The `#plist` statement now accepts integer and floating point literal values.

Editor

- Project
 - Resolved - The build settings window would occasionally display settings for standalone files, not the project.
 - Resolved - The build settings window would intermittently fail to close when clicking the Cancel or Save button.
- Document
 - Resolved - Choosing *Navigate/Jump to Definition* menu, with the `checkbox` or `radiobutton` keyword highlighted, would fail to open the *Button.incl* header.
 - In addition to header functions, text completion (start typing and hit 'esc' key) now shows records, `#defines`, systems and constants declarations.
 - Resolved - putting a comma immediately after the `timer` keyword would mess up indenting.

CocoaUI

- The `viewmenu` statement now adds Show Sidebar item to the menu when built for macOS 10.11+.
- SplitView sidebar support - A vertical splitview's left pane can now be made into a sidebar by calling *SplitViewMakeSidebar*.
- TextView dialog event, `_textViewCompletions`, is now hooked up. See FB Examples.
- The following WKWebsiteDataStore functions:
WKWebsiteDataStoreDefaultDataSore, WKWebsiteDataStoreNonPersistentDataStore & WKWebsiteDataStorePersistent
have been renamed to:
WKWebsiteDataStoreDefault, WKWebsiteDataStoreNonPersistent & WKWebsiteDataStoreIsPersistent

Note: This change will break existing code that calls these functions.
- Resolved - CImageByApplyingTransform second parameter had wrong type. Should have been CGAffineTransform.
- Other header updates and bug fixes.

FutureBasic Reference (FB Help)

- Resolved - SplitView statement *dividerStyle* and *vertical* parameters were in wrong order.

New Demos

- Views and Controls/
ColorWell/ColorWell Styles
ComboButton
TextView/Completions

Installation

- Now using a .dmg(was a .zip) image to help guide newcomers with the correct installation location.

20220806 FB 7.0.15

Translator

- Resolved - FB's build folder icon could be pixelated on non-retina Macs.
- Resolved - the `kill` keyword now highlights correctly in an editor window.
- Print/draw to window

The inset for a window's print/draw view has been removed in order to allow text and graphics to be positioned right up to the left and right edges of a window. Text 'indent' defaults to 3 pixels and the default value, for a specific window, can be adjusted by calling `WindowPrintViewSetTexttInset( wndTag, insetValue )`.

- Creating a new window with a missing *rect* param would previously position the window in the bottom left of the screen. With this release, the default is centered.

- New keywords

[combobutton](#). This is a new control introduced in macOS 13.0+ (see FutureBasic Reference and `ComboButton.incl` header).

[checkbox](#), [radiobutton](#). See `Button.incl` header and FutureBasic Reference for more info on these convenience keywords.

[intVal](#) and [dblVal](#). These are CF equivalents of the pascal [val&](#) and [val](#) functions (see FutureBasic Reference for more info).

[superview](#), [superviewend](#). These convenience keywords can reduce the amount of code needed when embedding views. See FutureBasic Reference for more info.

- The [left](#), [mid](#) and [right](#) functions can now be used in a select statement:

```
select left( string, 2 )
  case @"He"
  ....
end select
```

- The [colorwell](#) statement syntax has changed. Setting the *bordered* property of a colorwell is deprecated in macOS 13.0+ so the *bordered* parameter of FB's [colorwell](#) statement no longer sets the colorwell's border. On macOS 12 or earlier, a colorwell's border can be turned off by calling `ColorWellSetBordered( tag, NO )`. The colorwell's *bordered* parameter has been replaced by *style* (a macOS 13.0 feature).

Editor

- Resolved - A local function declaration using the 'clear' keyword would be omitted from the toolbar functions dropdown menu.
- Editor/Align/Declarations menu
Resolved - Declarations using the 'block' or 'static' keyword would be incorrectly aligned.
- Help menu - Rosetta Code item added. Choosing this item opens the Rosetta Code/FutureBasic website page in your browser.
- Build Settings window
Resolved - Changing BaseSDK value would occasionally leave Min deployment blank.

'Hide log window on suspend' has changed to 'Hide log window after build'.

NSLog

- When the NSLog window is key, selected text can now be copied to the clipboard by the keyboard shortcut ⌘-C. Also, ⌘-A can be used to select all text. Both these shortcuts work even when an Edit menu is not present.

CocoaUI

- Resolved - With Base SDK set to 11.0, macOS 11.0 APIs would not be available.
- Resolved - Apps calling the *TableViewSetHeaderFont* or *TableViewSetHeaderTextColor* functions would intermittently crash.
- Resolved - Subclassed views placed in a popover would fail to trigger `_viewMouseEntered`, `_viewMouseExited`, etc. dialog events.
- `CharacterSet.incl` - Function names fixed: *CharacterSetPunctuationSet*, *MutableCharacterSetPunctuationSet*. Note: these changes may break existing code that uses the old function names.
- `ColorWell.incl` - Style attribute introduced in macOS 13.0+. See the header and FutureBasic Reference for info.
- New header - `ComboButton.incl` (new control in macOS 13.0+). See FutureBasic Reference for info.
- Other header updates.

Translator

New shorthand 'getters' for CFArray and CFDictionary. See demos

```
obj = array[10]
obj = dict["@TheKey"]
```

New keywords:

[timerbegin](#), [timerend](#) - see demo and FB Reference for more info.

The deprecated dialog and menu functions have now been removed. Please use [on dialog fn](#) and [on menu fn](#) parameters instead.

The 'url' parameter of FB's open command no longer needs to be a variable pointer. The use of the '@' symbol is now deprecated.

e.g. this:

```
open "0", 1, @url
```

can now be:

```
open "0", 1, url
```

For the system function call:

- Reinstated the four constants (_scrnWidth, _scrnHeight, _aplActive, _sysVers)
- Updated FB Help (FutureBasic Reference) accordingly.
- Updated the _sysVers call to always return a 6 digit number (2 for major, 2 for minor, 2 for bug fix)

For the _constants file:

- Eliminated the file i/o by adding them to the symbol table in code.
- _constants file eliminated, see previous bullet
- _maxInt constant restored

build_temp directories are easier to spot with their new icon.

Editor

Project

Previously, a double-clicked item (file) would open in a third-party app *only* when it was in the 'Resources' group. With this release, files belonging to other apps can be placed in *any* group. If an owner app cannot be found for the double-clicked file, FB will attempt to open it as a document.

"Architecture" toolbar popup added. This optional toolbar item can be used to quickly view/set the current architecture for projects. The architecture item can be added to project window toolbars via the 'View/Customize Toolbar...' menu.

Resolved - The project contextual menu would incorrectly enable or disable some of its items.

'Show Constants' contextual menu item removed.

Document

"Architecture" toolbar popup added. Can be used to quickly view/set the current architecture for projects or standalone files. You can add this item to document window toolbars via the 'View/Customize Toolbar...' menu.

Resolved - 'Show in Finder' contextual menu item would occasionally reveal the wrong file.

Resolved - 'File Info' contextual menu item would sometimes show info for another file.

'File Info' contextual menu item now brings the document window to front.

'Show Constants' contextual menu item removed.

Resolved - The toolbar functions (fn) menu would occasionally contain a disabled item.

Command menu

Resolved - The 'Clean Build Folder' and/or 'Make Xcode Project' items would sometimes be disabled if a file belonging to a project was frontmost.

Resolved - The 'Clean Build Folder' and/or 'Make Xcode Project' items would be enabled if an unsaved document was frontmost and a project was open.

Clean Build Folder now quits the running app before deleting it.

'File Info' contextual menu item now brings the project window to front.

Debug menu

'Show Constants' item removed.

CocoaUI

Resolved - A SavePanelAccessoryView's buttons wouldn't be accessible via their tag values.

Vision framework added - see demo

New header - ISO8601DateFormatter.incl

Other header updates

New Demos

Collections/Array/
q Getter

Collections/Dictionary/
Shorthand Getter

Games/
Slide Puzzle

Graphics/Vision/
VNRecognizeTextRequest Demo

Miscellaneous/Key-ValueObserving
KVO Demo

Processes and Threads/Timer/
TimerBegin

20220425 FB 7.0.13

<http://www.brilorsoftware.com/fb/index.html> [you will need to reload the page if your browser cached a page mentioning a prior version]

Translator

#define type name max length increased to 100 chars (was 63).

The translator no longer quits the previously running app if the next build is for a different app. The prior behavior was any previous running app was terminated regardless.

Now creates a uniquely-named build_temp folder for each application (single source or project). Previously, a build_temp could be shared (most times when source was copied to the desktop) by multiple projects or single source apps and the build_temp's contents (because of the contamination by multiple apps potentially over a long period of time and potentially across FB releases) could be unreliable due to stale or out of date generated runtime. This would generate build/compile issues for FBers who hadn't noticed the situation. The build_temp folder name will now be build_temp plus the name of the project or source file. So "build_temp_MyFavoriteApp" is what you'll see. Its location hasn't changed.

New keyword - **static**

Previously, a static variable would need to be declared as follows:

```
begin globals
long sCounter// static
end globals
```

With this release, the above can be simply:

```
static long sCounter
```

Editor

Preferences Window

General tab 'Show file extension' option added. Allows display of filename extension in document window titlebars.

Note: the Finder's 'Hide extension' flag is determined by this setting at the time a file is created.

Import and Export buttons added to allow backup and restore of preference window settings.

Document

Save As Panel

'Show/Hide Extension' now appears in the options popup.

Extension popup added.

Hierarchical Menu

'Show File Extension' added. Toggles the display of file extensions in all document titlebars.

Resolved - Indent would fail if variable name, etc. on the same line as an 'if' statement contained "then", e.g. "authenticate".

Main Menu

Command/Clean Build Folder menu item now immediately deletes the build_temp folder and previously built app instead of moving them to the trash.

Jump to Line Window

Resolved - The Jump to Line window would remain on screen if it was visible when its document window closed.

Project Window

Hierarchical menu - 'Show File Extensions' added. Shows or hides file extensions in the file list of all project windows.

CocoaUI

LocalAuthentication framework added. See demo.

Speech framework added. See demo.

StoreKit framework added.

XML headers added. See demo.

MKPointAnnotation

MKPointAnnotationWithCoordinate function parameters have changed. The *title* and *subtitle* parameters have been removed and this function now takes just one parameter, *coordinate*. To create a MKPointAnnotation with *title* and *subtitle*, please use MKPointAnnotationWithTitle.

Two more Init functions added

MKPointAnnotationInit

MKPointAnnotationWithTitle(coordinate, title, subtitle)

Other header updates and additions.

Debug

NSLog

Resolved - The _NSLOG constant was still present but became unusable when NSLog was made into a framework.

New Demos

Security/LocalAuthentication/
LADemo

Sound/
AVAudioEngine

Strings and Text/
XMLParser

Strings and Text/Speech Recognition
Speech Recognition from File
Speech Recognition from Microphone

Views and Controls/View/
CameraView
AVCaptureView

Filters and Sorting/QuickSort/

Two new non-object QuickSorts: SortIntegers, QSBasic.

20220307 FB 7.0.12

<http://www.brilorsoftware.com/fb/index.html> [you will need to reload the page if your browser cached a page mentioning a prior version]

Translator

Building a command line tool(CLT) would fail with a rename error. Apparently nobody builds command line tools because this has been broken for a long time.

The rename now is done correctly. More information may be found below about CLT.

Editor

Changes to document and project toolbars may mean you need to reconfigure them to your preferences.

Resolved - Project and/or Document window toolbars would intermittently switch to regular size.

Resolved (we hope) - The editor would intermittently crash when pasting text.

Preferences Window

API warnings button added to General tab. This turns on/off checking for and displaying deprecated/unsupported APIs.

Comment block option added to Text Editing tab. This allows you to choose between slash-star (/ * */) comment block or double-slash (//) on each line. Default is slash-star.

CocoaUI

Resolved - default contextual menus for some nib views (TextView, TableView, etc.) would be disabled.

Various Header updates.

Building Command Line Tools (CLT)

Many FBers build only full GUI applications and might not have noticed its absence.

A command line tool, as the long-time FBers will remember, is a compiled binary that runs in the terminal app. All input and output is constrained to the Terminal.app window.

CLT examples are now compatible with FB7.

- Three CLT tools in FB_5_8_5_Examples/Non-CocoaUI demos/Command Line Tool
- Connect4 game in FB_5_8_5_Examples/Non-CocoaUI demos/Games/Connect4

Two more related items worth mentioning:

- Appendix J - CLT is sprinkled with inaccuracies and really needs an update, so check the examples first.
- CLT internals need an update. It's on the list. No implementation timeframe yet.

Examples

Updated SQLite demos from the 5.8.5 may be found in the FB Examples "Miscellaneous" folder

Screen capture to clipboard demo: FB Examples/Miscellaneous/Displays/ScreenCaptureToClipboard.fb

Screen resolution demo: FB Examples/Miscellaneous/Displays/ScreenResolution.fb

Phone Number Formatting: FB Examples/Data Formatting/Contacts/PhoneNumberFormat.fb

20220127 FB 7.0.11

<http://www.brilorsoftware.com/fb/index.html> [you will need to reload the page if your browser cached a page mentioning a prior version]

Translator

The [print](#) (to window) statement now prints CGRect, CGPoint, CGSize and CFRange values.

The [print](#) (to window) statement now supports print to *column*, *row*. See FutureBasic Reference for more info.

New keywords for drawing basic graphics to current output window. See FutureBasic Reference for more info.

[fill](#), [line](#), [oval](#), [pen](#), [rect](#)

An .iconset folder added to a project or included via the [include resources](#) statement is converted to an .icns file in the built app.

New keyword **#plist**. Adds a key/value entry to the built app's Info.plist. See FutureBasic Reference for more info.

Literals that span multiple lines can now be used to create an array or other collections (a new example is available - see below)

CocoaUI

WindowSetPrintViewFlipped(wndTag, flag) added to enable flipping the print view (or not).

Please note: this function is only for the special print view used by the 'print' to window command. Print views default to **flipped**.

Other non-print windows, which default to **unflipped**, can be flipped/unflipped with the standard function ViewSetFlipped(tag, flag).

Editor

Resolved - using the word **then** in a line comment (e.g. `// then`) would sometimes interfere with indenting.

Find in Project - search by *Regular Expression* option added.

Icon Assistant - the Icon Assistant window now has a choice of icon sizes to generate.

Resolved - closing a window after displaying the find bar would intermittently crash FB.

Resolved - FB's idler window would intermittently fail to hide or show when a document or project window was opened/closed.

Due to inconsistencies between Catalina, Big Sur and Monterey, FB's document and project toolbars were rebuilt. Their appearance has changed slightly but functionality is unchanged.

Headers

Numerous updates.

Examples

Mullti-line literal syntax example: FB Examples/Collections/Array/Create/Literal Syntax/MultilineLiterals.fb

Several excellent demos from the FB list have been added:

- MIDI examples: See: FB Examples/Sound/MidiPlayer and FB Examples/Sound/MidiKeyboard
- Examples of finding and using the color under the cursor. See: FB Examples/Views and Controls/View/ColorUnderMouse

20211124 FB 7.0.10

<http://www.brilorsoftware.com/fb/index.html> [you will need to reload the page if your browser cached a page mentioning a prior version]

Translator

New keyword **#build** is used to override Build Settings. See FutureBasic reference for more info.

The **textdraw** and **textstyle** keywords have now been replaced by **print** and **text**. See FutureBasic reference for more info.

New keyword **printf**. See FutureBasic reference for more info.

Resurrected keyword **cls**. **cls** clears any previously **printed** text in the current window. To reset text attributes to defaults, type the **text** keyword with no parameters

Editor

Project

Resolved - Choosing a function from the project item menu (option-click item) would fail to highlight the function.

Document

Resolved - Double-clicking a local function name in a project document would sometimes fail to highlight the function declaration.

Errors window

Resolved - Double-clicking an error/warning item would fail to highlight the item.

Resolved - The wrong filename would sometimes be displayed in the errors window.

CocoaUI

Event

Resolved - Calling EventKeyCode would sometimes crash.

Runtime

Fixed: Printing (print #1, gC) a container failed to build because PrintintContainerWnd() was missing from AppThings.c in build goodies.

Fixed: Passing empty container to instr\$ caused it to crash. Instr_Cntr_PStr() within bulid_goodies/Containers.c now returns zero for found position if container is empty.

Important Note on finding a container's length: Use *len\$* to retrieve a container's length.

Calling Container.c's internal functions (Len_Cntr and CntrRequestedSize) will fail (sometimes only silently).

Headers

Fixed: FB code wouldn't intercept TextView text selection changes. The dialog event, _testViewDidChangeSelection, was mismatched to its CocoaUI version.

New Demos

An example of the new PRINT to window and text keywords is available at: FB Examples/Strings and Text/Text/Print to window/Print To Window.fb

20211114 FB 7.0.9

<http://www.brilorsoftware.com/fb/index.html> [you will need to reload the page if your browser cached a page mentioning a prior version]

Translator

The old [mapview](#) keyword has been removed. Use [mkmapview](#) instead

New keywords:

[dispatchglobal](#), [dispatchmain](#), [dispatchend](#), [block](#) - see demo and FB Reference for more info

[textdraw](#), [textstyle](#) (require macOS 10.11+) - print text to the front window. These new keywords work similar to the old [print](#) and [text](#) commands - see demo and FB Reference for more info.

Gosub bug now displays an error instead of crashing FB. Hopefully, nobody is using Gosub.

As announced in 7.07 & 7.08 release notes, the translator doesn't report deprecation warnings in FB 7.0.9+. New editor features noted below should help.

'pi' now uses a math.h constant (it has a precision of 35 decimal digits). Before this change FB was using a Carbon header (fp.h) and Xcode flagged as deprecated.

Editor

Warning/Error indicator added to upper right of document window title bar.

Warns of deprecated/unsupported APIs and where feasible (like a new constant name) offers to fix the offending statement.

Statements unsupported by the selected (in Build Settings) SDK and/or Min Deployment are flagged as errors.

CocoaUI

Window - A window's 'Release When Closed' flag now defaults to YES for windows built with FB's [window](#) statement

NibWindow- Resolved. The 'Release When Closed' setting in the nib was ignored

Headers

Array	- more functions
AttributedString	- more functions
BezierPath	- convenience function added - BezierPathWithPolygonInRect - see demo
CISampler	- added function
Data	- more functions
Dictionary	- more functions
IndexSet	- more functions
LayoutManager	- more functions
Operation	- more functions
OrderedSet	- more functions
PopUpButton	- more functions
ScrollView	- more functions
Set	- more functions
SharingService	- removed harmless cast warning
StackView	- more functions
String	- more functions
TintColorConfiguration	- new header
Toolbar	- more functions

Window - more functions

New Demos

Graphics/BezierPath/BezierPathWithPolygonInRect
Processes and Threads/Grand Central Dispatch/Dispatch Keywords
Strings and Text/Text/Draw text in window/textdraw

FBHelp Updates

- Stop statement updated and deprecated code replaced
- End statement updated to more accurately describe its operation
- application page removed and all its constants moved to on appevent
- New: on appevent fn xxxxxx statement added to FBHelp
- New: Appendix N: Events provides an introduction to events with CocoaUI

20210915 FB 7.0.8

<http://www.brilorsoftware.com/fb/index.html> [you will need to reload the page if your browser cached a page mentioning a prior version]

Translator

The [alert](#) keyword msg param can now be either a CFStringRef or an ErrorRef.

One of the superceded/deprecated CocoaUI function warnings was incorrect. - fixed.
The translator will stop reporting these warnings in FB 7.0.9+. Please update your code. Last warning.

Lots of internal changes to purge old code.

CocoaUI

SharingServicePicker - added dialog events (see SharingService.incl header).

Editor

Resolved - Choosing 'Show Constants' in a project or document background window's contextual menu now correctly displays the constants for that project/document.

'Show in Finder' and 'File Info' main menu and project contextual menu items are now enabled for projects.

Resolved - Dated Comment menu item had lost its key equivalent (cmd-D).

Headers

Resolved - AVCaptureFileOutput leak plugged.
Cell.incl - more functions.
CharacterSet.incl - more functions.
TableView.incl - more functions

20210830 FB 7.0.7

<http://www.brilorsoftware.com/fb/index.html> [you will need to reload the page if your browser cached a page mentioning a prior version]

Translator

Pascal strings are completely removed from the translator's internal code. Translation times are improved as a result of this change.

Build settings now cached at the start of each translation.

Using Dim without a var type (i.e. 'dim x') is now an error. Vartype must be supplied (e.g. dim as short x)

Date/Tme keyword translation bug fixed.

Warnings are now given for superceded/deprecated CocoaUI functions (see *Headers/Mutable Objects* below).

Translation errors now displayed when building an app with the following UI elements on lower Base SDK than the one required.

[avcaptureview](#), [avplayerview](#), [wkwebview](#) (requires macOS 10.10+)
[avroutepickerview](#), [lplinkview](#), [switch](#) control (requires macOS 10.15+)

The old [webview](#) keyword has been removed. Use [wkwebview](#) instead.

New debugging keywords

sourcefilename	returns the source filename, containing this keyword, as a CFString
sourcelinum	returns the source line number, containing this keyword, as a long
sourcelinetext	returns the source line text, containing this keyword, as a CFString

Editor

- Main menu
- Editor/Align Assignments is now submenu - *Editor/Align/Assignments*
- Added items:
 - Editor/Align/Declarations*
 - Editor/Align/Assignments and Declarations*

Document

- Superceded/deprecated CocoaUI function names are now displayed as ~~strikethrough~~ text (see *Headers/Mutable Objects* below).
- Control clicking a superceded function name displays the replacement function name at the top of the context menu.
- Choosing this item replaces the old function name in the document.

Build settings

The lowest OS version in the 'Min deployment' popup is now 10.9.

New UI element statements:

[avcaptureview](#), [avroutepickerview](#) - See FutureBasic Reference for more info.

Headers

Array, NSAttributedString, CharacterSet, Data, Dictionary, IndexSet, OrderedSet, ParagraphStyle, Set, String
Functions that manipulate the mutable variations of the above objects now have "Mutable" prefix.

Examples:

Old	New
<i>ArrayAddObject</i>	<i>MutableArrayAddObject</i>
<i>AttributedStringSetFont</i>	<i>MutableAttributedStringSetFont</i>
<i>CharacterSetInit</i>	<i>MutableCharacterSetInit</i>
<i>DictionaryWithCapacity</i>	<i>MutableDictionaryWithCapacity</i>
<i>StringAppendFormat</i>	<i>MutableStringAppendFormat</i>

The old functions will continue to work for this release but **will be removed in the next release**, so please change them in your code at your earliest convenience.

Application - added custom functions
AppProperty, AppSetProperty, AppRemoveProperty, AppRemoveAllProperties

AudioToolbox - additions

AVKit and AVFoundation headers have now been rearranged with many duplicates removed.

UTType - new header

NSLog.incl
The debug macros (FUNCTION__, etc.) have been moved to their own header, *handy.incl* (Apple's name!).

handy.incl
New header (included as standard).

New Demos

Strings and Text/Anagrams/
Anagram Finder

New keywords

break

The **break** statement works in much the same way as the older **exit xxxx** statements but does not override any smart branching decisions the clang compiler may make.

continue

The **continue** statement works only within a loop, such as **for/next**, **do/until**, **while/wend**, and is somewhat like the **break** statement. Instead of forcing loop termination like **break**, it skips the remainder of the current loop iteration and resumes the loop at the beginning.

Fixed: the Errors/Warnings window would truncate longer messages.

[ANN] FB's 'on timer' syntax, even though it translates without error messages(but fails to compile), calls older/ unsupported Carbon Events and isn't usable.

FB 7.0.6 will display an error message if 'on timer' is used.

CocoaUI provides 'on timer' functionality for those needing it.

An example may be found at: FB Examples/Processes and Threads/Timer/AppSetTimer.fb

Resurrecting 'on timer' syntax is probably possible but other higher priorities take precedence right now. On list discussion welcome.

Translation

A key translation function is now much faster resulting in reduced (up to 50%) translation times.

The function is called hundreds of times during a build, so efficiency improvements really pay off.

Replacing the pascal string processing enabled the speed improvements and that project is ongoing.

Per the announcement in July 2020 (FB 6.0.13) and March 2021(FB 7.0.5), all the ancient constants(many go back to ZBasic) have been removed.

Add or change your constants as needed if you see errors.

Headers and code updated to use POSIX file path instead of the outdated HFS style (kCFURLHFSPathStyle) using colons (ThisDirectory:thisFileName) .

Change doesn't impact FBer code but it would have broken the translator. Apple marked it deprecated in OS X 10.9.

NSLog

Resolved - Clicking the 'i' (info) button on NSLog's settings popover would crash the app.

Editor

Idler Window

'New File' button now has a pull down menu: New File, New File from Clipboard.

Document

The optional toolbar 'Find' icon now displays the document find bar instead of the Find in Headers panel.

Resolved - 'Revert to Saved' menu item would display divider lines in their old format string ('~'1).

An alert is now displayed when attempting to amend a Header file. Options are Continue and Cancel.

Jump To Line Window

Line number field now only accepts digits.

Line number field now grows and shrinks with content making it easier to grab and drag window.

Project

Choosing Find in Project or Find and Replace in Project now makes the 'Find' field key view.

'Find' button added to Find in Project window.

Constants Window

Double-clicking a User-defined constant name now opens the document containing the constant and scrolls down to it.

Editor Menu

Align Assignments item added. Choosing this menu item aligns the first occurrence of an equals ('=') sign in each line of the selected text.

View Menu

Icon Assistant item added. Displays a window where image files can be dropped to generate an iconset folder and icns file.

Generated files are placed in the same directory as the image file dropped on the Icon Assistant window.

Headers

AnimationContext

AnimationContextRunAnimationGroupWithHandler - callback function parameters have changed - see header for examples

AVKit

More headers and functions

View

ViewAddToolTipRect now works as intended (see DynamicToolTips demo)

WKWebpagePreferences - new header

Other headers - added functions

CharacterSet - Added function

TlBx DictionaryServices.incl - Added functions

Per the announcement in FB 7.0.5, the following include files have been removed. Use equivalent CocoaUI functions or FB keywords instead.

Util_FileManager.incl

Util_PathUtilities.incl

Util_RunningApplications.incl

Util_String.incl

Util_URL.incl

Util_UserDefaults.incl

New Demos

Views and Controls/

ScrollView/SynchronizedScrollViews

View/DynamicToolTips

20210331 FB 7.0.5

<http://www.brilorsoftware.com/fb/index.html> [you will need to reload the page if your browser cached a page mentioning a prior version]

General

Resolved - Turning off NSLog, via the Debug/NSLog menu, in a project would only remove NSLog.incl if it was in the project's Headers group.

Resolved - Apps including a private framework with a path containing unix escape characters would fail to compile.

Resolved - Enclosing [openpanel](#) or [savepanel](#) keyword parameters in the optional parentheses would throw error.

The [instr](#) keyword is now for CFStrings only and can no longer be used with pascal strings. Pascal strings must now use [instr\\$](#) instead. See FutureBasic Reference for more info.

Note: This change will break existing code.

Select block now handles CFStrings

```
select ( myCFString )
  case @"Charlie"
    // ...
  case @"Foxtrot"
    // ...
  case @"Sierra"
    // ...
end select
```

NSLog

Additional keyboard shortcuts when NSLog window is key:

cmd-F shows find bar

cmd-S displays save log panel

NSLog calls can now be made from a background thread.

Translation

Resolved - PSstr() crashed because CopyDbIToString() didn't return a valid string (i.e. it had already been released). Can occur for any parsing and translation calling PSstr()

Note: PSStr() supports legacy pascal strings (typically for FB's 'str\$' keyword) and is part of the old pascal string stack logic.

WARNING: Pascal string support will eventually be removed from FB.
The warnings have been there for years, so this shouldn't be a surprise.
You have time to convert now. We won't have much sympathy for those who wait and complain when they are "surprised" by the change.

Headers

The following headers are deprecated and will be removed in the next release. Please use equivalent CocoaUI functions or FB keywords instead.

- Util_FileManager.incl
- Util_PathUtilities.incl
- Util_RunningApplications.incl
- Util_String.incl
- Util_URL.incl
- Util_UserDefaults.incl

CocoaUI

- Application - more functions
- Calendar - more functions
- ScrollView - added function
- TableView - more functions
- View - convenience functions added for working with radio button groups: ViewSelectedButton, ViewSelectedButtonTag, ViewSelectedButtonTitle
- ViewController - more functions

New Demos

- Graphics/CA/
 - CAEmitter/Fire
 - CAEmitter/Earth Moon and Stars
 - CAEmitter/Longitude Demo
 - CALayer/CALayerMask
 - CALayer/Clock Demo
 - CALayer/CALayer Demo
- Processes and Threads/Timer
 - Let's Dance
- Views and Controls/TableView/Header Font and Colors/
 - TableViewHeaderColors 1
 - TableViewHeaderColors 2
 - TableViewHeaderFont

20210202 FB 7.0.4
<http://www.brilorsoftware.com/FB/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

New keywords

To assist in the transition from Pascal strings, a few extra string handling keywords have been added. See FutureBasic Reference for more info.

<u>CFString</u>	<u>Pascal</u>
bin	
cvi	cvi\$
date	
hex	
lcase	

left
len
mid
mki
oct
right
space
str
time
ucase

Note 1: The pascal string versions of **cvi** and **len** have now become **cvi\$** and **len\$**. This change will break existing code.

Note 2: If your app uses any of the new keywords as variable names or function names, they'll need to be changed in your code.

Note 3: The dollar sign '\$' suffix indicates the keyword either accepts and/or returns a pascal string. Conversely, keywords in this group without a '\$' suffix indicate CFString usage only.

STR\$ Keyword Change

- **str\$** was completely rewritten to address old(2009) deprecated code. It continues to convert a double to a pascal string and its usage is unchanged.
However, there are some important differences which **str\$** users ***should review carefully.***
 1. **gFBFloatMaxDigits** still defaults to a value of 10 but It now specifies the maximum number of decimal(i.e. fractional) digits the resulting string should contain.
It does not influence digits in the integer portion. The global's description in 'FB Help appendix D - Numeric Expressions' was not clear on this distinction.
 2. **str\$** will remove trailing decimal zeros. So if the the double is '123.980000', it will return '123.98'. The same is true for leading zeros on the integer portion of the double.
 3. If **gFBFloatMaxDigits** is set to 3, '-8001.102' will deliver '-8001.102' but if **gFBFloatMaxDigits** is set to 2, then it only looks at two decimal digits (the '.10' and not the 2).
Since '-8001.10' has a trailing zero it would return a string of '-8001.1'.
 4. **str\$** also rounds decimal digits, as appropriate, based on **gFBFloatMaxDigits** and the double's value.
With **gFBFloatMaxDigits** set to 3, '-83.123789' results in a pascal string of '-83.124' because the '7' rounds the 3 to 4 and the global specifies three decimal digits.
 5. If **str\$** is used with the global unchanged (i.e. left at the default of 10 digits), the results should be the same as the previous version of **str\$**.
Old behavior of prefixing an output string space for non-negative numbers is retained.
 6. FBers are responsible for setting or not setting **gFBFloatMaxDigits** appropriately in their code. The following table shows output differences with **gFBFloatMaxDigits** as shown. The 10 digit output also reflects the original input data (sans any leading space).

New STR\$ with 3 digits	Old STR\$ with 3 digits	New STR\$ with 10 digits	Old STR\$ with 10 digits
- 3 4 . 1 2 3	- 3 4 . 1 2	- 3 4 . 1 2 3 4 5 6 7 8	- 3 4 . 1 2 3 4 5 6 7 8
2 8 4 9	2849	2 8 4 9	2 8 4 9
- 1 . 6 5	- 1 . 6 5	- 1 . 6 5	- 1 . 6 5
1 2 . 3 4 5	1 2 . 3 4 5	1 2 . 3 4 5	1 2 . 3 4 5
- 1 2 3 4 . 2 2	- 1 2 3 4 . 2	- 1 2 3 4 . 2 2	- 1 2 3 4 . 2 2
- 8 0 0 1 . 1 0 2	- 8 0 0 1 . 1	- 8 0 0 1 . 1 0 2	- 8 0 0 1 . 1 0 2
- 8 3 . 1 2 4	- 8 3 . 1 2	- 8 3 . 1 2 3 7 8 9	- 8 3 . 1 2 3 7 8 9

NSLog

- **NSLog** has been rewritten and is now a framework. Its standard usage within FB hasn't changed. Just include "NSLog.incl" and you're good.
- Many ancillary functions, such as **NSLogSetShowsDate**, have been dropped and most settings must now be done via **NSLog**'s settings popover on log window.
The dropped functions are:
NSLogSetShowsDate
NSLogSetShowsTime
NSLogSetShowsAppName

NSLogSetShowsProcessID
NSLogRestoreDefaultFont
NSLogIsSuppressed
NSLogSuppress
NSLogSetFrame
NSLogFrame
NSLogShow
NSLogHide
NSLogSetFloatingWindow
NSLogWindow
NSLogSetBecomesKeyOnlyIfNeeded
NSLogSetTerminateAppWhenWindowClosed
NSLogSetTextWrap
NSLogSettingsButtonSetHidden
NSLogWrapTextButtonSetHidden
NSLogFindButtonSetHidden
NSLogSaveButtonSetHidden
NSLogClearButtonSetHidden
NSLogButtonsSetHidden
NSLogSetForwardToConsole
NSLogBeginEditing
NSLogEndEditing
NSLogStartProgressIndicator
NSLogStopProgressIndicator

- The default font can no longer be set programmatically, and must be done manually via the font panel in the settings popover. The NSLogSetFont call now sets a temporary font.
- The log window can now be closed (cmd-w) even when a File > Close menu item does not exist.
- Please note a special usage case: Using NSLog.incl in an Xcode project requires an NSLog framework copy in the project...
 1. Make Xcode Project.
 2. Option-drag(to make a copy) NSLog.framework to the Xcode project's "General/Frameworks, Libraries, & Embedded Content" section.
 3. Ensure the Embed popup is set to either "Embed & Sign" or "Embed Without Signing".

Help menu

String Format Specifiers are vital part of string usage, so a link to them on Apple's web page with that same title has been added.

Project

Automate Build Number. A new feature that automatically increases the prior build number by one. Especially helpful meeting Apple's MAS submission requirements.

To enable automatic build number increment, a project must have an Info.plist. The steps are:

Choose 'Automate Build Number' in the project context menu and the automate build number sheet will be displayed.

Choose a build number type (integer, decimal or hexadecimal - your choice).

The initial build number, the increment value, and the number of digits can be set in this sheet.

Each build or run of the project will increment the "CFBundleVersion" value in the Info.plist.

Document

New document files now default to "fb" extension. This can be overridden but must be one of the standard FB extensions (fb, main, glbl, incl, bas).

Examples window

- Double clicking an item in the results table now scrolls to and highlights the first occurrence of the search string in the opened document.
- Resolved: After using the search field on the Examples window to create a 'found' list of source files, the 'Show in Finder' contextual menu command fails.

CocoaUI

- NibWindow

Resolved - A window with toolbar would fail to set a toolbar item's _toolbarItemClick dialog action if the item did not have an attached view.

- Function names beginning `XxxxIsEqual` have now been standardised. For example, `StringIsEqualToString` has now become `StringIsEqual`.

Headers affected are: [Array](#), [Data](#), [Date](#), [DateInterval](#), [OrderedSet](#), [Set](#), [SKNode](#), [String](#), [Value](#).

Note: The old functions remain in the headers so as not to break existing code.

- `TableView`

For convenience, any table columns that have not been given an identifier are automatically set to the column header title (if any) when the tableview is loaded from the nib.

Translation

• A build will now skip subsequent steps (copy resources, compile, link etc.) when translation errors are found. In other words, an error-free translation is required for a compile/link.

Translation *warnings* do not impact the compile/link steps. This should save time for big projects with lots of errors.

FB Examples

All stand-alone FB demo files have now been given an ".fb" extension.

New Unfixed Bugs Since last release

- The FB example at: FB Examples/Strings and Text/HTML does not respond correctly when system Dark Mode is set on.
- Embedded `/*` in user code math calculation causes all source past that point to show as commented. Workaround is to put a space between the slash and asterisk.

20210113 FB 7.0.3

<http://www.brilorsoftware.com/FB/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

Menu

- Show Examples item added to View menu (see Examples Window below)

Examples window

- The Examples window can be used to display and search demos in the FB Examples folder.

To show the Examples window, choose View > Examples from the main menu.

If the examples folder cannot be found, you will be asked to locate it. The folder location is remembered for next launch.

CocoaUI

- `TextView`

New dialog event constant `_textViewShouldChangeTextInRange`

Translation and Build

- Rebuilds with the same build_temp directory skip compiles for unchanged user files. Implemented in FB 7.0.2
- FB 7.0.3 implements the above feature for the FB runtime.

N.B.

If Source-file combining is used, keep in mind a small change in one file forces the recompile of all files that are combined. Setting Source-file combining to "Off" avoids this.

20210105 FB 7.0.2

<http://www.brilorsoftware.com/FB/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

New keyword

The ***available*** keyword is used to check availability in a conditional statement. Example:

```
if ( available(11.0) )
    // use macOS 11.0 APIs
else
    // alternative code for earlier versions of macOS
end if
```

Build Log window

- Filter field added to bottom content border

- Circular progress indicator now displayed during build
- Contextual menu added
 - Save As... save the log as an rtf file
 - Clear Log

Constants and File Info popovers

- The detached windows for these popovers now become active immediately when detached

Project

- A project window frame and its file window frames are now remembered between launches
- On opening a project file, item names are matched to same case as their filenames

Translation and Build

- Rebuilds with the same build_temp skip unchanged files.
FB 7 releases broke this previously-working (FB 6) feature. We're happy it's back!
Note: Works for user files but not for the translated runtime file.
- Known issue: All FB 7 single architecture builds are slightly slower compared to FB 6.0.16.
Brian thinks he knows the cause. Bernie is readying Starbucks delivery to keep him focused.
- Translator code cleanup and minor bug fixes.
One fix: the built app's icon now appears without the  symbol at build completion.
- Future possibilities:
Speed improvements for universal builds(parallel builds) and FB runtime (skip if not changed).
These are still just ideas, so don't get excited yet.

20201231 FB 7.0.1

<http://www.brilorsoftware.com/FB/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

Bug Fix

Labels (as used in 'goto') failed to convert from C strings to CFStrings resulting in a nil pointer and crash when subsequent code tried to use it.

Errors file

Errors encountered during build are no longer saved to an Errors file. The Errors and Warnings window displays errors as before.

Menu

Clean Build Folder now closes the Build Log window, if it's showing

Errors window

Save button added to errors window bottom border

20201229 FB 7.0

<http://www.brilorsoftware.com/FB/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

FBtoC

FBtoC and the Editor are now combined. There is no longer a separate FBtoC app - Farewell FBtoC!
Universal runs on Intel and M1 Macs

ToDo

- Update the translator to accept passing of 'NS' types instead (or in addition to) of 'CF' types (i.e function header could accept: local fn Dolt (str as NSString *) vs. local fn Dolt(str as CFStringRef)
- Run 'Make Xcode Project' on background thread

Bugs

• Simple apps (for example with no UI) built with optimize and dead code stripping are significantly larger than the same code built in Xcode

For example, a single source Brian uses, when built with FB , is over 500k but 150k in Xcode.

- When a string containing backslash double-quote is used somewhere in an 'if' statement, FBtoC hangs
if (fn CFStringHasSuffix(s, @"\\"")) // hangs when backslash double-quote is used in an 'if' statement

workaround - don't used backslash double-quote anywhere in an 'if' statement

dim as Boolean flag

flag = fn CFStringHasSuffix(s, @"\")

if (flag)

- Inline assignment fails for shift operator

long k = (500 + 0) >> 1 // this fails translation

workaround #1 - put all in parenthesis

long k = ((500 + 0) >> 1)

workaround #2 - put on separate lines

long k

k = (500 + 0) >> 1

- Inline assignment fails for certain CG constants such as CGPointZero, CGSizeZero, CGRectZero

CGPoint pt1 = CGPointZero // fails

workaround 1:

CGPoint pt2

pt2 = CGPointZero

workaround 2:

CGPoint pt3 = {0,0}

Editor

- FB7 now builds Apple Silicon, Intel or Universal apps. See Build Settings

General

- On FutureBasic launch, it now checks if Xcode exists in the Applications directory. If it doesn't, an alert is thrown up with a button that links to the Xcode page in the App Store

Menu

- Make Xcode Project item added to Command menu

Document

- Unified Toolbar item added to contextual menu. Choosing this item will toggle all open document window toolbar styles between expanding and unified

Note: This menu item is only available when running on macOS 11.0+ BigSur

Project

- Unified Toolbar item added to contextual menu. Choosing this item will toggle all open project window toolbar styles between expanding and unified

Note: This menu item is only available when running on macOS 11.0+ BigSur

Build Settings window

- New 'Architecture' popup button with items: "Apple Silicon", "Intel", "Universal"

Note that Apple Silicon and Universal items are only visible when Base SDK is set to 11.0+

CocoaUI

- Nib Menu - When creating a nib menu, the About, Hide, Quit and Help items default to "NewApplication". With this release, these item titles are automatically changed to the app name on launch

20201107 FB 6.0.16

<http://www.brilorsoftware.com/fb/index.html> [you will need to reload the page if your browser cached a page mentioning a prior version]

Editor

- Command/Help Book menu
 - Create item added - Creates a help book shell
 - Help book menu items are now only enabled for projects
- Project
 - Resolved - project window would sometimes forget its previous size and location
 - Find and Replace

- Replace All is now carried out on a background thread and the Find/Replace window cannot be closed during a Replace All operation
- Holding option key whilst expanding/collapsing an item now expands/collapses all items

- Document
 - Resolved - one or two predefined types would not highlight
 - Resolved - when editor launched by double-clicking a file, some toolboxes would not be highlighted

CocoaUI

- General
 - Resolved - some calls would crash when an app built with base SDK higher than the OS of the Mac running the app.
- Toolbar
 - More functions
 - ToolbarSelectedItemIdentifier
 - ToolbarSetSelectedItemIdentifier
 - ToolbarSelectedItem
 - ToolbarSetSelectedItem
 - ToolbarSetSelectableItemIdentifiers
- ToolbarItem
 - Extra function - ToolbarItemSetSelectable
- Window
 - Extra function - WindowSetKeyViewLoop

New demos

- EventHandling/KeyViewLoop
 - WindowSetKeyViewLoop

FBtoC

- Obsolete TBAliasStatement, TroffStatement and TronStatement: Not supported for many years but now turned off.
- Fixed: Memory overwrite when strings used as labels exceeded 64 characters caused FBtoC crash.
- Pascal String Stack: Effort to eliminate continues.
- Symbol table search speed tweak. An older function character by character string compare was replaced with one call to `CFStringCompare()`

Bugs(and in some cases workarounds)

- Simple apps (for example with no UI) built with optimize and dead code stripping are significantly larger than the same code built in Xcode

Status: awaiting investigation

- When a string containing backslash double-quote is used somewhere in an 'if' statement, FBtoC hangs


```
if ( fn CFStringHasSuffix( s, @"\\"" ) ) // hangs when backslash double-quote is used in an 'if' statement
```

workaround - don't used backslash double-quote anywhere in an 'if' statement

dim as Boolean flag

flag = fn CFStringHasSuffix(s, @"\\"")

if (flag)

Status: awaiting investigation

- Inline assignment fails for shift operator


```
long k = ( 500 + 0 ) >> 1 // this fails translation
```

workaround #1 - put all in parenthesis

```
long k = ( ( 500 + 0 ) >> 1 )
```

workaround #2 - put on separate lines

long k

k = (500 + 0) >> 1

Status: awaiting investigation

- Inline assignment fails for certain CG constants such as CGPointZero, CGSizeZero, CGRectZero


```
CGPoint pt1 = CGPointZero // fails
```


workaround 1:
CGPoint pt2
pt2 = CGPointZero

workaround 2:
CGPoint pt3 = {0,0}
Status: awaiting investigation

20200824 FB 6.0.15

<http://www.brilorsoftware.com/FB/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

FBtoC

- Translation capabilities are no longer dependent on the editor. Any editor capable of communicating with FBtoC will work.
- Syntax

- Inline C

- A new convenience keyword, `cln`, allows a single line of C code to be inserted. This keyword avoids having to enclose a line in a BeginCCode...EndC block. e.g.

```
-  
    cln [btn setMaxAcceleratorLevel:4];
```

is equivalent to

```
BeginCCode  
    [btn setMaxAcceleratorLevel:4];  
EndC
```

Please note: the FBtoC translator does not check 'cln' C code lines for correct C/Objective-C syntax; that responsibility lies with the programmer just as it is with the block style. (i.e. BeginCCode/endC)

- Inline record assignment

- With FB6.0.14 came inline assignment for the more common records (CGRect, CGSize, CGPoint and CFRange). With this release, *all* records (predefined and user-defined) can use inline assignment

- The parentheses around a record assignment became a problem when the new 'all records' feature was implemented. Setting a field value with a parenthesised expression would cause ((may)hem)

```
MyRecord myRec = ( (x+10)*20, 30 )    // << nope!
```

so with this release braces (not parentheses) are required around record assignments

```
MyRecord myRec = { (x+10)*20, 30 }    // << yep
```

- Subrecords - With the exception of CGRect, subrecord fields must also be enclosed in braces

```
begin record ItemPrice  
    float  buy  
    float  sell  
end record
```

```
begin record ItemInfo  
    long    itemID  
    long    qty  
    ItemPrice price  
    long    order  
end record
```

```
ItemInfo info = { 54321, 23, { 12.34, 17.53 }, 9 }
```

Editor

- Preferences

- 'Show bookmarks in functions menu' checkbox added to the Text Editing pane of the Preferences window. When checked, bookmarks are added to the document toolbar functions menu (see notes in Project and Document below)

- Project

- Find and replace window

- Find/Replace popup button added. The replace field and buttons are now only visible when the popup's 'Replace' item is selected

- Clicking the Replace All button now throws up a confirmation alert
- The search word is now highlighted in each line in the results table
- Find menu
 - Find in Project... (⇧⌘F) now always shows the Find window in 'Find' mode
 - Find and Replace in Project... (⇧⇧⌘F) added. Displays the Find window in Replace mode
- Toolbar Find in Project item (magnifying glass) shows the Find window in the mode it was last displayed. Holding option key whilst clicking the magnifying glass will display the window in Replace mode.
- Project window functions menu (opt-click item)
 - In previous releases, the function names were always sorted. With this release, sort/unsort is determined by document functions toolbar item setting
 - This menu can now optionally show bookmarks as well as functions (see Preferences above)
- Document
 - The toolbar functions menu can now optionally show bookmarks as well as functions (see Preferences above). When the menu is unsorted, an item separator is inserted before each bookmark. Separators do not appear in a sorted menu
 - Choosing an Editor > Font Size item (⌘+, ⌘-) now briefly pops up the current font size

CocoaUI

- PDFAnnotation - extra function
- PDFAnnotationUtilities - fixes for NS/CFString compiler warnings
-

20200813 FB 6.0.14

<http://www.brilorsoftware.com/FB/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

FBtoC

- 'long if' produces a deprecation warning. Please update code to use 'if' instead.

Editor

- Project
 - Organization Identifier field added to save panel. Entering a value in this field generates an Info.plist and sets its CFBundleIdentifier value. The Info.plist is added to the project's Resources group
 - Project find/replace
As this is a new feature, please make a backup of your project before doing a project-wide find/replace

Usage

1. Click on the project toolbar magnifying glass or choose Find > Find in Project... (⇧⌘F)
2. Select your search options in the two popup menus
3. Enter the search text and hit return
4. Results are displayed in a table below the fields
5. Double click a table item to open the document
6. To replace items
 - a) Enter the replacement text (if any) in the Replace field
 - b) Select one or more items in the table (drag select, shift-click, cmd-click, Edit > Select All, etc.)
 - c) Hit the Replace button

N.B. Any items in a collapsed group will not be replaced
7. Hitting the Replace All button replaces all items, even those in collapsed groups

Important Find/Replace Usage Note:

Project find/replace currently finds and replaces text *in files*, **not text in open documents**, so be sure to save and close any open documents that are part of the project *before* doing project find/replace

- Document
 - File menu Save All (⇧⌘S) item added
 - Resolved - Some actions would corrupt the contents of the general pasteboard
 - Divider line and bookmark special characters are now converted to their old character sequence in the saved file. This prevents the divider lines/bookmarks being lost when editing the file in an old editor.
- Dim statement enhancements (**dim as** syntax only)

- Declaration and assignment can be done at the same time

```
dim as long      value = 12
dim as CFStringRef s1 = @"One", s2 = @"Two", s3 = @"Three", s4 = @"O'Leary"
dim as CGRect    r = fn CGRectMake( 14, 13, 110, 32 )
```
- 'dim as' keywords are now optional

```
long      state
CFStringRef string
double    d1 = 123.456, d2 = 987.654
BOOL      flag = YES
```
- CGRect, CGPoint, CGSize and CFRange structures can be initialised like this

```
CGRect    r = ( 0, 0, 360, 270 )
CGPoint    pt = ( 120, 50 )
CGSize     size = ( 187, 42 )
CFRange    range = ( NSNotFound, 0 )
```

CocoaUI

- Array - More functions

New Demos

- Shows new 'dim as' syntax: FB_6_0_14_Examples/Miscellaneous

20200731FB 6.0.13

<http://www.brilorsoftware.com/FB/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

FBtoC

Keywords removed

- 'cocoa' & 'cocoaalert' keywords, deprecated in August 2019, now generate a build error. The shortened version without the 'cocoa' prefix should be used (i.e. 'alert') instead

Pre-Defined Constants file change

- _constants file's values, updated to decimal instead of hexadecimal, allows faster FBtoC processing
- New editor functionality (see below) helps manage both pre-defined and user-defined constant use within your project.
- Most of the pre-defined constants (over 5300 of them) in the _constants file are obsolete, no longer applicable and in some cases provide incorrect/misleading results.
- They date back to pre- OS X days (QuickDraw). We strongly suggest reviewing your code's usage of these pre-defined constants and find alternatives such as your own constants.
- Most of these pre-defined constants from the _constants file will be removed in a future release, so please plan accordingly and take advantage of the new editor functionality.

Pascal String Stack

- Effort to eliminate commences.

NSLog

Resolved - Settings would sometimes fail to be stored in the correct preferences plist

'Appearance' popup added to settings. Allows the NSLog window to be displayed in the user's preferred appearance (light or dark). Available macOS 10.14+

Editor

Preferences window

- General pane
 - 'Appearance' popup added. This popup gives the user the option to work with the Editor in light mode or dark mode, irrespective of system setting. Available macOS 10.14+
- Fonts & Colors pane
 - Line numbers checkbox now defaults to checked
 - Line numbers checkbox no longer shows/hides line numbers in documents. To hide/show line numbers in documents, use the Editor menu item or the document toolbar Line Numbers item (if configured)
 - Divider line width and color options added

File Menu

- File Info - File attributes list now include line count and character count

Editor Menu

- There is now just one divider line style. The default divider line style is one pixel wide and text color. Opening a document or pasting text into a document will convert any old divider line styles to the one style

Command Menu

- The 'Trash build_temp' item renamed to 'Clean Build Folder.' In addition to trashing the build_temp folder, this action now trashes an Errors file and the built app linked to the front document or project

Debug Menu

- Resolved - Switching NSLog on or off would resave unchanged project documents. With this release, only *changed* documents are resaved
- 'Show Constants' item added. Displays all user-defined and predefined constants used in the front project or document

Document

- Resolved issues
 - Switching to dark mode would intermittently fail to switch already open documents to that appearance
 - After pasting text into a document, text insertion would often fail to appear in the correct location
 - When a new version of the editor is launched by double-clicking a document file, the document would appear (unformatted) behind the EULA window. Now, the document window is not displayed until the EULA window has been closed
 - During live resize of a document window, revealed text would appear unformatted until the mouse released
- Toolbar
 - Resolved - In FB 6.0.12, the line number label displayed a bad range when only one line selected
 - The FN menu and Function name toolbar items are now combined. Note this change may reset the toolbar to default items on first launch of this editor version
 - The function menu's "Sort" item title no longer changes its name but displays as either ticked (✓) or unticked
 - When no selected function in the document, the function menu now displays "Functions" instead of three dots (...)
 - The character count in the line number label looked too ugly (IMO) so has been removed. Character information is now displayed as a tooltip (hover over the toolbar line number label)
- Documents now respond to click-through. i.e. clicking on an inactive document now activates the text view with one click instead of clicking first to make the window active and then clicking again in the text view
- Divider line
 - Historically, FB uses a special character sequence to flag divider lines and bookmarks, e.g. '~'1. As this multi-character sequence required a nasty workaround to silence glyph warnings, divider lines and bookmarks are now flagged by a single, invisible character (\2, \3 respectively). Saved files will contain these new characters and old character sequences will be converted to the new single character when opening a file.
However... for backwards compatibility:
 - Old character sequences will be converted to the new single character when pasting text into a document
 - Text copied from a document converts the new divider line character to the old 1 pixel character sequence (~'1) on the clipboard. Useful for sharing code with those who are running an earlier version of FB
 - Divider line lengths now grow/shrink with the window width or text line length
 - A single click on a divider line highlights the whole line
- Contextual menu - 'Show Constants' item added

Errors window

- New filter field at the bottom of the window's border limits the text display to errors containing the entered text

Project window

- Contextual menu - 'Show Constants' item added

New exception handling keywords (variables with same names as keywords will need updating)

- try, catch, finally, throw. See FutureBasic Reference and demos.
- User variables matching the new keyword names will be flagged as errors.
- N.B. Editor-based feature.

Usage Notes for the Editor's new Show Constants Window

1. The "User-Defined" column shows the constants added by the user to the document/project.
2. The "Predefined" column shows constants the project/document uses from the _constants file & the FB Headers.
3. If a predefined constant is colored red, a floating popover indicates it is "not a valid constant" (see #4 for definition of "invalid")
4. Except for a small subset of valid constants(see below), most of the _constants file's constants are invalid.

Invalid means: obsolete, no longer applicable and might provide incorrect/misleading results (maxInt is an example of the latter).

Valid Constant List from _constants (still evolving)

btnclick
definedincarbon
definedincocoa
definedincruntime
false
fbtoc
fnferr
itemid
menuid
nil
noerr
nslog
sysvers
toprinter
true
zblack
zblue
zcyan
zgreen
zmagenta
zred
ztrue
zwhite
zyellow

CocoaUI

Cell - More functions

Dictionary - More functions

Exception - New header

Expression - Extra function

Number - Extra functions for handling 32-bit numbers: NumberWithInt, NumberWithUnsignedInt, NumberIntValue, NumberUnsignedIntValue

ViewController - More functions

Window

- New function, *WindowPerformWindowDragWithEvent*. A new demo (see below) uses this function
- New dialog event, *_windowCanBecomeKey*. A new demo (see below) uses this event

New Demos

Debug/try_catch_finally_throw examples

Filters and Sorting/Expression

- Expression Demo
- Expression Demo (with exception handler)

Views and Controls/

- ImageView/DropImage
- PopUpButton/Variable Width/Variable Width Popup

Windows and Panels/Window/Custom/Custom

New FB 5.8.5 Demos

MDSArray (multi-dimensional array) demo was updated. See: FB_5_8_5_Examples/Non-CocoaUI demos/CoreFoundation/MDSArray 1.0.6

New Util_String.incl header includes a replacement for FB's bin\$ that returns a CFStringRef instead of a pascal string. See: FB_5_8_5_Examples/Non-CocoaUI demos/CoreFoundation/CoreFoundation/Strings and Numbers/Strings/Demo of Replacement String functions/BinStringDemo.bas

20200703FB 6.0.12

<http://www.brilorsoftware.com/FB/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

FBtoC

Build Settings

- Min deployment and Base SDK popups now handle the macOS 11.0 SDK correctly.

NSLog

- Adding a `\b` or `\t` to the end of the format string puts the next NSLog message on the same line.
- Appending the format string with `\a` pauses the app and displays an alert. This is identical to calling `NSLogAlert`.
- The log font (set by calling *NSFontSetFont*) is now remembered between launches and after clearing the log.
- Code clean up.

Editor

Project

- Min deployment and Base SDK popups in Build Settings now handle the macOS 11.0 SDK correctly.
- Existing files in the project window are no longer highlighted during drop operation.
- Project window contextual menu rearranged with new items: Open in New Tab, Delete.
Context menu now shows any links found in clicked document. Shows first word in the line with the link, or the link itself if no word found.
- With two or more projects open, and build settings sheet displayed in one of the projects, a click on another project's settings cog would switch the sheet to the clicked window. Now, only one build settings can be displayed at any one time.

Document

- Improved scrolling and text editing response with large files.
- BeginC block text now retains its leading space characters after Re-Indent or Trim Whitespace.
- C code text coloring has been dropped. C code is now displayed the same as other text.
- Contextual menu now contains any links found in document. Shows first word in the line or the link itself if no word found.
- Standalone file build settings (FBtoC settings) are now displayed in the editor. Therefore, the FBtoC app is no longer launched to display its settings.
- Lines/chars toolbar label now displays total number of characters in the file.

CocoaUI

PopUpButton

- A nib popup button stopped issuing `_btnClick` dialog event in FB 6.0.11. Fixed.

MenuToolbarItem

- The second parameter of `MenuToolbarItemSetMenu` can now take a menu index or `CocoaMenuRef`.

StatusItem

- *StatusItemSetMenuIndex* function has been removed and its operation combined with *StatusItemSetMenu*. *StatusItemSetMenu* now accepts a menu index or a menu ref. This change will break existing apps that call *StatusItemSetMenuIndex*.

UserDefaults

- `UserDefaultsSetFont` now stores the font's font descriptor rather than the font object itself (recommended by Apple).
- More functions

Widget menus

- Widget menu handling has been updated.
 - If appropriate for the widget (`PopUpButton`, etc.), choosing a menu item triggers a `_btnClick` dialog event and the item chosen can be got by calling `PopUpButtonIndexOfSelectedItem( tag )`, etc.
 - Submenus (hierarchical menus) in the widget menu always trigger an on menu event. i.e. they do not generate a `_btnClick` event.

Window

- New dialog event constant `_windowShouldPopUpDocumentPathMenu`.

New Demos

- Miscellaneous/
ObjectProperty

- Views and Controls/PopUpButton/
Submenu

20200622 FB 6.0.11

<http://www.brilorsoftware.com/FB/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

Editor

Debug Menu

- The way the NSLog menu item works has changed.
In addition to adding/removing the NSLog.incl header to/from the front project or document, turning the NSLog item off (unchecked) comments any code lines starting with "NSLog" with double-slash-squiggle "//~". Turning the NSLog item ON (checked) removes these special comment characters from any code lines beginning with "//~NSLog".
Notes:
 - NSLog calls that don't appear at the start of a line are ignored. For example, the NSLog call in this line is ignored: **if (a == b) then NSLog(@"equal")**
 - Existing NSLog statements commented by the regular double-slash comment "//NSLog" are ignored.

Help Menu

- 'FutureBasic Help' item renamed 'FutureBasic Reference'.

Document

- Bug fix. Comment blocks would lose their space character indent when the insertion location was next to the slash character in the start comment block (/*).
- The text insertion location is now correctly restored after Re-Indent or Trim Whitespace action.

Show Headers Window

- Contextual menus for the CocoaUI headers now contain links to Apple's online documentation.

CocoaUI

Alert

- More functions.

AppleEventDescriptor

- More functions.

Array

- *ArrayShuffledArray* function has now been moved to its correct header: GameplayKit/GKRandomSource.incl. In previous versions, to gain access to this function required including the GK framework (*include library "GameplayKit"*). With this release, we need to include a header instead: *include "Tlbx GameplayKit.incl"*. This change will break existing apps that call this function.

Box

- Omitting or passing NULL in the title param of the box statement now sets the box title position to NSNoTitle. Note: setting the box title to NSNoTitle does not change the box coordinates but *visually* changes its height.
- More functions.

Button

- More functions.

CATransform3D

- More functions.

ComparisonPredicate

- Extra function.

Data

- Extra function.

Geometry

- More functions.

Menu

- *MenuAddItem* and *MenuInsertItem* have been renamed to *MenuAddItemWithTitle* and *MenuInsertItemWithTitle* respectively. Also the new functions return a MenuItemRef so require the 'fn' keyword before the function name. These changes will break existing apps that call these functions.
- *MenuSetSubmenu* now also removes any action assigned to the parent item.
- More functions.

MenuItem

- More functions.

Operation

- *OperationSetCompletionFunction* now takes an extra parameter, *userData* (can be NULL). This change will break existing apps that call this function.

Also, the user completion function now has two optional parameters:

```
void local fn MyCompletionFunction( ref as OperationRef, userData as ptr )
    // ...
end fn
```

OperationQueue

- The OperationQueue.incl header has been removed and its functions transferred to Operation.incl.

PathCell

- More functions.

PathControl

- The path control now triggers its dialog events by default so the *PathControlSetDelegate* function has been removed. This change will break existing apps that call this function.
- New dialog event constant *_pathControlDoubleAction*.

PopUpButton

- *PopUpButtonSetMenuIndex* function has been removed and its operation combined with *PopUpButtonSetMenu*. *PopUpButtonSetMenu* now accepts a menu index or a menu ref. This change will break existing apps that call *PopUpButtonSetMenuIndex*.

PrintOperation

- New on dialog event constant *_printOperationDidRun*. Triggered on completion of the print operation after calling *PrintOperationRunModal*. See header for more information.

Set

- More functions.

TableView

- More functions.
- Table data can now hold CFNumbers. Any CFNumber objects found in the table data are automatically entered in the text field cell.
- New dialog event constant *_tableViewCellTextDidEndEditing*. Called when an editable text field has done editing.
- Reorder single row has been available for a while but with this release *multiple* rows can be reordered. See demo.
- Drag copy row implemented. This feature is off by default but can be turned on by calling the convenience function *TableViewSetAllowsDragCopy*.
- The behaviour of drag reorder has changed. Previously, when NSTableViewDropOn was enabled (default) dragging an item *down* would move the item immediately *below* the highlighted item. With this release, dragging up or down always puts the item *above* the highlighted item.
- Although the table view's drop operation can be handled in the *_tableViewValidateDrop* dialog event, a new convenience function (*TableViewSetDropOperation*) will do the same thing without the need to intercept the dialog event.
- Drag drops are now animated.

TabView

- More functions.

TabViewItem

- New header.

TextCheckingController

- New header.

TextField

- More functions.

TextStorage

- More functions.

UserDefaults

- More functions.

Value

- More functions.
- CATransform3D value function moved to the CATransform3D.incl header.
- Geometry value functions moved to Geometry.incl.
- Range value functions moved to Range.incl.

View

- More functions.

General

- Nib view menu indexes would occasionally fail to be identified. Fixed.

New Demos

- Collections/Array/Sort/
Array/Sort/Array Sort
- Views and Controls/TableView/
InsertRemove

20200506 FB 6.0.10

<http://www.brilorsoftware.com/FB/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

FBtoC

build_goodies files removed

- EditFields.c
- FilesDollarFunction.c
- Prefs.c
- SystemDirectoryCopyURL.h
- SystemDirectoryCopyURL.m
- Util_FileDirectory.h
- Util_FileDirectory.c

Headers removed

- Subs Help tags.incl
- Subs Prefs File.incl
- Tlbx Aliases.incl
- Rntm Completion.incl
- Subs Files.incl

Make Xcode Project Options Reduced

- Make Xcode Project's dual options are reduced to one "without Stripping" option and renamed to simply: "Make Xcode Project" in FBtoC's file menu. Changed to run on a background thread.

Editor

Project

- Double-clicking a folder in Resources group now opens the folder in the Finder
- Bug fix. The optimization label would not update when build settings window loaded.

Document

- Hitting return in a comment block would remove leading whitespace from entered line. Fixed.
- Lines with '#' prefix are no longer indented.
- On re-indent, lines in comment blocks would march to the right. Fixed.
- Re-indented comment block lines now retain any leading space characters.

Help Menu

- Files or apps dropped in the Application Support/FutureBasic/Help Menu Items/ directory now appear in the Help menu immediately. i.e. there's no need to quit and relaunch FutureBasic.

FutureBasic Help
Additions

CocoaUI
Animation

- New header

Application

- Extra function

Bundle

- For convenience, *NULL* can be used instead of *fn BundleMain* in BundleXxxx functions. e.g.
dict = fn BundleInfoDictionary(NULL)
is equivalent to
dict = fn BundleInfoDictionary(fn BundleMain)

Button

- More functions

CAAnimation

- More functions

FileHandle

- Bug fix. FileHandleSeekToEnd would crash when second param NULL.

Responder

- Extra function

String

- Extra function
- Function name corrections: StringByReplacingOccurrencesOfString, StringByReplacingOccurrencesOfStringWithOptions
Note: these corrections may break existing apps.

TableView

- The dialog events: _tableViewSelectionShouldChange, _tableViewShouldSelectRow and tableViewShouldSelectTableColumn all default to YES but can now be manipulated by calling DialogEventSetBool(NO). Also, in _tableViewShouldSelectRow, the row number can be got with: row = fn DialogEventLong.
- It is now possible to add custom views to one or more columns. See demos.

TextFinder

- More functions

TextView

- More functions

UndoManager

- Extra function

URL

- More functions

View

- More dialog events
- More functions

VisualEffectView

- More material types (macOS 10.14+)

Window

- More functions

Workspace

- More functions
- WorkspaceOpenConfiguration class added (macOS 10.15+)

General

- Programmer error alert window
 - Now runs on main thread to avoid crash when triggered from background thread.
 - 'Quit' button added. Useful for when the alert is triggered in a loop.

New Demos

- Miscellaneous/
 - UndoManager
- Views and Controls/SKView/
 - Animated Fruit
 - Bouncing Robot
 - Flowers
 - GIF Animation
- Views and Controls/TableView/Custom Views/
 - Button
 - Checkbox
 - DatePicker
 - MultipleViews
 - PopUpButton
- Windows and Panels/
 - Shake

20200316 FB 6.0.9

<http://www.brilorsoftware.com/FB/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

FBtoC

Bug fix: A block comment (i.e. those beginning with `/*` and ending with `*/`) where an individual line(defined as all the characters up to the first end-of-line character -ascii 13) within the block exceeds 255 characters would cause unpredictable behavior - usually a loop/hang. Pascal-dependent code and pascal- influenced limitations were replaced to remove constraints.

Note: The maximum line length inside a block comment is now the same as any other (non-comment) FB source line(1500 characters per line). To clarify: Overall block comment length is still unlimited but each line within the block must be less than the maximum per line.

Editor

Editor Menu

- This new menu contains *Divider Line*, *Dated Comment*, *Comment Block*, *Trim Whitespace*, *Re-Indent* (moved from the Edit menu) and *Hide Line Numbers*, *Wrap Text* (moved from Views menu).
It also contains new items: *Font Size/Increase*, *Font Size/Decrease*, *Create Code Snippet*, *Edit Snippet Name*.

Application Support - Help Menu Items

- Files or apps can now be added to the Editor's Help menu by dropping them in `~/Library/Application Support/FutureBasic/Help Menu Items` directory. Added items will appear in the Help menu on the next launch of the Editor.

Code Snippets

- Code snippets are displayed under the Snippets tab in the Library panel.
Note: Currently only supports files but a later version of FB will allow folders too.

To create a new code snippet:

either
Drag a text file on to the snippets panel
or
Drag text from a document window to the snippets panel
or
Select text in a document and choose Editor>Create Code Snippet

To edit a snippet name:

Ctrl-click a snippet name in the snippets panel and choose Edit Snippet Name from the contextual menu.
or
Select an item in the snippets panel and choose Editor>Edit Snippet Name

Project

- QuickLook preview enabled. Press space bar to show selected items in QLPreview.

CocoaUI

AppleScript

- Error dictionarykeys added. See header.

Application

- *AppEnumerateWindows* params and its callback function have changed.
 - Note: This change may break existing apps that use this function. Please see header.

Array

- *ArrayOfObjectPassingTest* & *ArrayEnumerateObjects* function params and their callback functions have changed.
 - Note: These changes may break existing apps that use these functions. Please see header.

AVKit

- More functions. See headers.

CALayer

- More functions. See header.

Color

- More functions. See header.

Control

- More functions. See header.
- A new custom function, *ControlSetActionID(controlTag, actionID)*, can be used to separate radio button groups. Previously, multiple groups would have to be embedded in separate superviews. Note: The actionID value must be in the range 1-10.

CIFilter

- *CIFilterWithNameAndInputParamters* function name (missing 'e') corrected to *CIFilterWithNameAndInputParameters*.
 - Note: This change may break existing apps that use this function. Please see header.

CoreImage

- More headers.

Data

- More functions.

DialogEvent

- New constant, `_viewMenuForEvent`. Call *DialogEventSetMenu(cocoaMenuRef)* to set the menu.
- The `_viewUpdateTrackingAreas` event is now hooked up. Note: When responding to this event, call *DialogEventSetBool(YES)* to let the runtime know you've handled it.

Dictionary

- *DictionaryEnumerateKeysAndObjects*, *DictionaryKeysOfEntriesPassingTest* & *DictionaryKeysWithOptionsOfEntriesPassingTest* params and their callback functions have changed.
 - Note: This change may break existing apps that use these functions. Please see header.

FileHandle

- Many name changes and parameter changes which may break existing apps that use these functions. Please see header.

FileManager

- Extra function.

Font

- Extra function.

FontDescriptor

- Extra function.

FoundationErrors

- More constants.

JSONSerialization

- Additional writing option.

ListFormatter

- New header and demo.

MenuItem

- Attempting to configure a menu item that doesn't exist would occasionally hangs - fixed.

Number

- More functions. See header.

Object

- Extra function. See header.

Operation

- Extra function.

QuickLookThumbnailing

- New framework. Requires: macOS 10.15+ and Tlbx QuickLookThumbnailing.incl.

QuickLookUI

- New headers and demo.

PersonNameComponents & PersonNameComponentsFormatter

- New headers and demo.

RelativeDateTimeFormatter

- New header and demo.

Responder

- More functions. See header.

Screen

- More functions. See header.

SpellChecker

- New headers.

Storyboard

- New headers and demo (the beginnings of storyboard handling).

String

- *StringEnumerateLines*, *StringEnumerateSubstringsInRange* & *StringEnumerateLinguisticTags* functions were broken. Fixed.

TableView

- Dialog event `_tableViewDidAddRowView`. The row number and `TableRowViewRef` can now be got with *DialogEventLong* and *DialogEventView* respectively. See demo: Row Background Color.
- Subclassed tableview dialog event, `_tableViewAcceptsFirstResponder`, now defaults to `_true`.
- Unused dialog event constants removed.
- The *TableViewSelectedRowIndexes* function now correctly returns an `IndexSetRef`, rather than a `CFArrayRef`. This change may break existing apps that use this function.

TextView

- Extra function. See header.

ToolbarItem

- More functions. See header.

TouchBar

- New headers and demo.

TrackingArea

- *TrackingAreaWithRect* now returns a valid *TrackingAreaRef*.

URLError

- More constants. See header.

View

- Subclassed views no longer automatically respond to *_viewMouseEntered*, *_viewMouseExited* & *_viewMouseMoved* events. In order to receive these events, you must now call *ViewInitTrackingArea(viewTag)* at build time.

WebKit

- More functions. See headers.

General

- Most subclassed widgets can now respond to *_viewMouseEntered*, *_viewMouseExited* & *_viewMouseMoved* events. Note: in order to receive these events a new custom function, *ViewInitTrackingArea(tag)*, must be called on the view at build time.
- Callback functions which optionally returned *_true* (YES) to stop enumeration now no longer return a value but take an extra param, *stp* as *^BOOL*. For example, the callback function for *AppEnumerateWindows...*

was:

```
local fn MyEnumerateWindows( wndTag as NSInteger, userData as ptr )
    // ...
end fn // = YES    // stop enumeration
```

now:

```
void local fn MyEnumerateWindow( wndTag as NSInteger, stp as ^BOOL, userData as ptr )
    // ...
    /*stp = YES      // stop enumeration
end fn
```

Objects affected:

Application, Array, AttributedString, Calendar, Contacts, Dictionary, DraggingInfo, DraggingSession, EventKit, LinguisticTagger, NaturalLanguage, OrderedSet, RegularExpression, SpriteKit, String

New demos

- Collections/Array/
 - ArrayByApplyingDifference
- Collections/Dictionary/
 - KeysOfEntriesPassingTest
- Data Formatting/
 - ListFormatter
 - PersonNameComponents
- Dates and Times/
 - RelativeDateTimeFormatter
- Graphics/
 - BezierPath/ImageWithDrawingHandler
 - Core Image/... lots of lovely demos - thanks Ken!
 - JuliaSet Viewer
- Windows and Panels/
 - QLPreviewPanel
 - TouchBar
 - Storyboard
- Views and Controls/TableView/
 - Row Background Color

20200114 FB 6.0.8

<http://www.brilorsoftware.com/FB/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

FBtoC

Compiler options

- The '-Wno-unguarded-availability' compiler option is now issued on every build by default.
- This compiler option eliminates the need to set min deployment to 10.13+ in order to take advantage of features introduced in macOS 10.13+.
- This option can be turned off by entering: '-Wunguarded-availability' in compiler options. However, this will give hundreds of warnings when building apps.
- Non-MacRoman characters, such as 'č', in a Pascal or CFString literal/constant would generate failing CFSTR() macro code. Fixed

Editor

TouchBar support

- A MacBook Pro's TouchBar now displays relevant controls for most of the Editor's windows.
-
- Note: Those FBers who don't have a MacBook Pro with TouchBar can get an idea of what it looks like by...
 1. Launching Xcode
 2. Choosing Window>Show Touch Bar (cmd-shift-8)
 3. Switching back to FutureBasic

Application menu

- License Agreement item added. Opens license agreement in Preview.

File menu

- File Info item added - displays file attributes in detachable popover.

Help menu

- FBtoC Website bad link fixed.
- Release Notes item added.
- FutureBasic Help
- **subclass** keyword page updated.
- Bad links to FBtoC website fixed.

Project

- Toolbar would intermittently forget configuration - fixed.
- File Info contextual menu item and File Info toolbar item added - displays file attributes in detachable popover.

Document

- File Info contextual menu item and File Info toolbar item added - displays file attributes in detachable popover.
- Text completion enabled for toolbox functions. Start typing function name then hit 'esc' key to display list of functions that begin with word.
- Non-MacRoman characters, such as 'č', in a Pascal or CFString literal/constant would cause document save failure leaving the source file empty. All now saved as UTF-8.

Objects Panel

- Filter field added to object panel's content border.

CocoaUI

Application

- NSApp global variable added.
- More functions.

BackgroundActivityScheduler

- New header and demo. Requires macOS 10.10+.

Button

- Extra function.

ByteCountFormatter

- More functions.

ColorSampler

- New header and demo. Requires macOS 10.15+.

CompoundPredicate

- More functions.

Control

- More functions.

Error

- More functions.

ImageRep

- More functions.

LinkPresentation

- New framework. See headers and demo. Requires "Tlhx LinkPresentation.incl" header and macOS 10.15+

Measurement, MeasurementFormatter Unit

- New headers and demos. Requires macOS 10.12+.

MenuIndex

- All MenuIndexXxxx functions now optionally accept a MenuIndexRef for the itemIndex param. Note: if a MenuIndexRef is entered, the menuIndex param value is ignored.

NaturalLanguage

- New framework. See headers and demos. Requires "Tlhx NaturalLanguage.incl" header and macOS 10.14+

OrderedSet

- More functions.

SearchField

- Convenience functions added: SearchFieldSetSearchButtonImage, SearchFieldSetCancelButtonImage. See demo.

StackView

- StackView added. See header and demos. Requires macOS 10.9+.

Switch

- Switch control added. See header and demos. Requires macOS 10.15+.

View

- More functions.

Demos

New

- Data Formatting/
 - Measurement
- Collections/
 - OrderedSet
- Strings and Text/
 - NaturalLanguage
- Views and Controls/
 - LinkPresentation
 - SearchField ButtonImage
 - StackView
 - Switch
- Windows and Panels/
 - ColorSampler

Minor cleanups.

- Removed all "Requires -m64 compiler options" comments.
- Changed links from the old FBtoC website to the new one.
- All images for Graphics/DrawThreePartImage and Graphics/DrawNinePartImage enclosed in "images" folders.
- Views and Controls/TabView _tabViewDidSelectTabViewItem added to on dialog function.
- Fixed a couple of comment typos.

20191219 FB 6.0.7

<http://www.brilorsoftware.com/FB/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

Editor

- Fixed a bug where Find in Project would sometimes hang.
- Drag and drop failed to update FB's project file in some circumstances.
- Key words ending in a dollar sign, '\$', are now highlighted in an editor document.
- An app's Help Book can be indexed by choosing Command/Index Help Book
 - Choosing that menu item indexes the help book if the following are true:
 - Project/document folder contains an Info.plist
 - The Info.plist contains a help book folder
 - The help book folder exists
- The following bugs were fixed after first 6.0.7 release:
 - Header include file specified within project window was incorrectly treated as a resource
 - Reopening a project which has been set to a Base SDK that no longer exists leaves the Base SDK popup blank
- * dialog(0) and dialog(ev) functionality restored

```
i.e. dim as long ev, id, wnd
      ev    = dialog( 0 )
      id    = dialog( ev )
      wnd   = dialog( -1 )
      select ( ev )
```

CocoaUI

- Clicking TabView now triggers _tabViewDidSelectTabViewItem dialog event instead of _btnClick.
- AVPlayerView subviews now trigger these dialog events:
 - _viewPerformDragOperation
 - _viewDraggingEntered
 - _viewDraggingExited dialog event

Runtime & Other

- Calls using FSRefs removed from header file, Tlhx LSOpen.incl
- Clean up of General.c, Runtime.h
- FixSDK's broken import/export in Catalina now fixed. FixSDK is a separate download on the FBtoC web site
- Internal FBtoC changes to omit the include files EditFields.c, FilesDollarFunction.c
- AppThings.c reduced to only the live code it uses. Most of the conditional compiles removed.
- AppThings.c updated to restore PRINT # support

20191002 FB 6.0.6

<http://www.brilorsoftware.com/FB/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

Overview

FB 6.0.6 adds some nice new features, continues the effort to streamline and simplify FB, and continues the effort to remove FB's runtime Carbon code. More work remains on the last two items.

Please note the warnings for the User Interface items.

FBtoC 5.8.10, runtime and general internals

Sixty five FB headers were removed because they rely on 32-bit code and/or other obsolete technologies. build_goodies also updated. This process is ongoing.

FB continues to link against the Carbon.framework because Apple's recent Catalina beta includes some bits of Carbon. There is little to no build time speed penalty for linking against the Carbon.framework with most FBers using CocoaUI. Please note: FB no longer includes Apple's Carbon headers, so your code might need them if Carbon calls are made. One way to include the Carbon headers is shown below. Linking against Carbon will be removed when Apple discontinues its inclusion in the OS.

```
beginCDeclaration
#import <Carbon/Carbon.h>
endC
```

User Interface related

File handling - OSPanel

OSPanelOpen, OSPanelSave and their ancillary functions are now deprecated. Use CocoaUI's openpanel and savepanel instead. As usual, deprecation is a prelude to removal, so update your code accordingly.

Printing - PrintInfo

In earlier releases, PrintInfo functions would configure the shared PrintInfo object. With this release, all PrintInfo functions require a PrintInfoRef paramater. **Warning: this change may break existing apps**

Views and Controls -

ViewController - The ViewControllerView function now returns a ViewRef, rather than a tag value. **Warning: this change may break existing apps**

PathControl - The PathControl widget is now available. See PathControl in FBHelp, PathControl.incl header and the PathControl Demo

Editor related

Menu Bar

Now has a Debug menu which currently contains just one item, 'Include NSLog'. Choosing this item adds NSLog.incl to the top group of the front project or inserts an include statement at the top of the front document

User Keywords Panel

The User Keywords Panel now has a contextual menu containing one item, 'Show in Finder'. Choosing this item reveals the UserKeywords.plist file in the Application Support folder

Project

Option-clicking a project window file item now displays a list of the file's functions (three-finger trackpad touch does same thing)

Show Headers Folder (on View menu)

Option-clicking a header reveals all its functions. Choosing a function from the list opens the header to that function and highlights it. (three-finger trackpad touch does same thing)

Examples

New example- Views and Controls/PathControl

Updated 'Non-CocoaUI' demos are:

- Non-CocoaUI demos/CoreFoundation/MDArray(multi-dimensional)
- Non-CocoaUI demos/CoreFoundation/CoreFoundation/CFTThings in FB console
- Non-CocoaUI demos/Games/Connect4

20190830 FB 6.0.5
<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

Overview

This release reflects a major effort to consolidate the CocoaUI Reference and the FB Help Reference. Older FB Help Reference entries were updated only when necessary. Help is now html-based.

As noted in the 6.0.4 Explanatory Comments (see release notes for 6.0.4), FB 6.0.5 continues the ongoing effort to streamline and simplify FB. Many of FB 6.0.4's cautionary notes still apply and FBers should take action with their source. A review of those release notes is recommend with particular attention to the plan to remove linking against the Carbon.framework .

FBtoC 5.8.9

- Shows deprecation warnings for 'cocoa' keyword and 'cocoaalert' statement/function
- 'window close' and 'window output' statements now work with Cocoa windows
- Simpler macro declaration

Editor 6.0.5

Menus

- Find menu moved out of Edit menu to menu bar
- Menus rearranged

Project

- New Project template updated
- Fixed: Internal drag/drop had stopped working in 6.0.3 and 6.0.4

Document

- More keywords supported in ctrl-click word

Dragging a Headers file to a document window now adds an include statement in the document (e.g. **include** "NSLog.incl"), rather than the contents of the header.

Preferences

Options to notify of build success or failure

Find in Headers window (opened via the 'Find' menu)

'Find in Apple Headers' added. Note: the older 'Search Apple Headers' app is retired, redundant and no longer included in bundle

Help

HTML-based FBHelp updated and included in editor
FB Reference.pdf file removed from help menu
CocoaUI Reference.pdf removed from help menu

General

Application Support directory name changed from "FBEditor" to "FutureBasic"
Path for System.framework as used in Open "C" corrected
Runtime bug fix corrects Open "C" failure

Minor bug fixes and improvements

CocoaUI

Changes that may break existing apps

- 1. CNContactPicker dialog event constant names changed. Now have 'cn' prefix such as _cnContactPickerDidSelectContact, etc.
- 2. CLLocationManager dialog event constant names changed. Now have 'cl' prefix such as _clLocationManagerDidUpdateLocations, etc.

The **cocoa** keyword is no longer required and will be removed soon

- ~~cocoa~~ box
- ~~cocoa~~ button
- ~~cocoa~~ menu
- ~~cocoa~~ window

The **cocoaalert** statement/function is deprecated and will be removed soon. Use **alert** instead

Headers changed

Graphics.incl

New headers

SpeechRecognizer.incl
SpeechSynthesizer.incl
Stream.incl (includes InputStream + OutputStream)
UbiquitousKeyValueStore.incl
UserActivity.incl

20190808 FB 6.0.4
<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

FB 6.0.0+ requires OS X 10.13+

N.B. Please read the following carefully *BEFORE* updating to this release.

Overview and Explanatory Comments

This release is the first phase of an effort to streamline and simplify FB. Goals/Accomplishments:

- 1. Focus FB to create only 64-bit apps and not 32-bit. This release creates only 64-bit binaries.
- 2. Focusing on 64-bit only allows a significant reduction in the number of runtime headers FBtoC processes that are no longer necessary.
 - a) FB Runtime headers were changed to exclude old Carbon code and in some cases (serial I/O) update code for 64-bit Cocoa.
 - b) As noted above, this is the first phase. There is more work in the runtime but this is a good start.
 - c} Internal testing shows better build times after reducing the includes processed. Makes sense right?

3. Reduce complexity and simplify where possible.

- a) it is no longer necessary to specify -m64 -DDECARBONATE. FBtoC adds those options automatically
- b) FBtoC finds the path to the developer folder automatically, so that popup has been removed from build settings
- c) it is no longer necessary to include Tlxb CocoaUI.incl or code 'cocoainit'.
- d) other improvements are planned but not implemented.

4. **Utility Headers N.B.** This release removes many of the 'Util_xxxx.incl' headers (see list below).

These headers have been discouraged since CocoaUI's introduction (because of the duplication and incompatibilities) but many FBers might still be using them. Many of the removed 'Util_xxxx.incl' headers have incompatibilities with the current runtime, so keep this in mind if you elect option '4b' below.

Your options, with most preferred first, are:

- a) Switch to CocoaUI calls and remove your project's references to the old 'Util_xxxx.incl' headers. Recommended approach.
- b) Make copies of the 'Util_xxxx.incl' headers your project uses and store them in your project or the FB User Libraries directory (i.e. /Users/yourUserID/Library/Application Support/FutureBasic/User Libraries). FBer is responsible for resolving any incompatibilities (see note above) with existing runtime.
- c) Use the prior FB release (6.0.3)

5. This release continues to link against the Carbon framework to allow some old structures (FB records) to build without errors but that will change in the next release or so. Some examples of these records are: EventRecord, RGBColor, Rect, DateTimeRec, LongDateRec, FSRef, AEDesc and others. If you're still using those or any Carbon-defined records, consider making your own local copies.

FBtoC 5.8.8

Build settings

- Path to Developer Folder popup removed
- -m64 -DDECARBONATE compiler options no longer required

Util_xxxxxx.incl Headers removed:

Util_CE.incl
Util_CFPrefs.incl
Util_CFPrefsCFStrings.incl
Util_CFUN.incl
Util_Dictionary.incl
Util_EUTCKeyFilter.incl
Util_FileDirectory.incl
Util_Files.incl
Util_Folders.incl
Util_MouseTracking.incl
Util_Nib.incl
Util_NibCFStrings.incl
Util_Number.incl
Util_Sound.incl
Util_Toolbar.incl
Util_UI.incl
Util_Workspace.incl

FutureBasic 6.0.4

General

- Builds only 64-bit apps. 32-bit app builds removed and not supported.
- "Tlxb CocoaUI.incl" header included automatically
- 'cocoainit' statement no longer required
- Bug fixes and tweaks

20190722FB 6.0.3

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

FB 6.0.0+ requires OS X 10.13+

FBtoC 5.8.7

Maximum function (i.e. FB function 'fn') parameters increased to 40 (Brian)

Cosmetic: log window scrollview border removed

FutureBasic

Preferences

Redesigned window. Prefs items split into three tabs: General, Fonts & Colors, Text Editing
Document background color option added

Document

Click in toolbar fn name label goes to first line of function only when click is within the function name label text

Click in toolbar line number label brings up goto line number window only when click is within the line number label text

Jump to Definition now does not select word but briefly shows find indicator and puts cursor at beginning of word

Fix: cmd-double-click calls Jump to Definition, rather than its own code

Line numbers and text wrap items removed from toolbar default set

Project

Fix: dragging a header file to the project would copy the header file to the project directory

Drag to project now shows 'Link' icon for Header files, 'Copy' icon for other files

File icon size increased for Regular font size

Headers window

Drag from Headers window to project enabled

Errors window

Fix: double-click now opens *.m file reliably (if appropriate)

License agreement window

Redesigned

FBEditor User Guide

Updated

Other minor bug fixes and tweaks

20190713 FB 6.0.2

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

FB 6.0.0+ requires OS X 10.13+

FB 6.0.2 release notes:

- Drag (move) and option-drag (copy) selection within document
- Drag copy selection between documents
- Document contextual menu for one-word widget statement show template syntax (select to paste)
- Find Definition for one-word widget statements opens widget header file
- Show Headers Folder displays Headers directory and CocoaUI directory in tabbed window
- Bug fixes and improvements

CocoaUI and FBtoC are unchanged for this release and carry forward from FB 6.0.1

20190702 FB 6.0.1

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached a page mentioning a prior version]

FB 6.0.0+ requires OS X 10.13+

FB6.0.1 release notes:

Project

- double-clicking an FB file item now opens the file in FB6, rather than FB5.
- option-clicking a group's expand/collapse triangle expands/collapses all groups

- two font sizes choices (small & regular) - set via preferences or project contextual menu
- find in project now has its own panel (project find & replace will be in next release)

Document

- Edit menu 'copy' removes leading tab characters and replaces divider line attachments with text equivalent before putting on pasteboard

CocoaUI changes that may break existing apps

TextView

The textview statement's *subclass* parameter has been removed. Syntax is now:

textview tag, rect, scrollViewTag, textContainerRef, wndTag

N.B. to subclass a textview, use the **subclass** keyword (see CocoaUI Reference)

View

The view statement's *subclass* parameter has been removed. Syntax is now:

view tag, rect, wndTag

N.B. to subclass a view, use the **subclass** keyword (see CocoaUI Reference)

CocoaUI Additions and changes

MenuItem

additions

TextView

parameter removed

The textview statement's *subclass* parameter has been removed. Use the subclass keyword instead (see CocoaUI Reference).

View

parameter removed

The view statement's *subclass* parameter has been removed. Use the subclass keyword instead (see CocoaUI Reference).

20190618 FB 6.0.0

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.8.5]

FB 6.0.0 requires OS X 10.13+

[01] The FBtoC website has been revamped/reorganized with links to 64-bit editors

20190610 FB 5.8.5

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.8.4]

FB 5.8.5 requires OS X 10.7+ for CocoaUI apps. Non-CocoaUI requires 10.6

CocoaUI changes/additions and new demos

Changes that may break existing apps

OutlineView

Function name changed from OutlineItemAddChild to OutlineItemAddChildWithString

Tag value upper limit

Window and widget tag values must now be less than 1000000

Additions and changes

Application
additions

Array
additions

AVPlayer
addition

AVPlayerView
additions

Dictionary
additions

IndexSet
additions

NumberFormatter
additions

OutlineView
additions and changes

SplitView
additions

SpriteKit
new
See SpriteKit.pdf documentation in FB's Help menu and SKView demos

String
additions

subclass
new keyword
When placed next to a window or widget statement, a subclass of the window/widget is created. Please read about the *subclass* statement in CocoaUI documentation.

TableViewCellView
new

TableColumn
new

TableHeaderView
new

TableView
additions

TextView
parameter deprecation
The textview statement's *subclass* parameter will be removed in the next release. Use the 'subclass' statement instead

Toolbar
change

ToolbarItem
additions

View
parameter deprecation
The view statement's *subclass* parameter will be removed in the next release. Use the 'subclass' statement instead

Window

addition

New demos

Views and Controls/OutlineView/DragDrop/

DragDrop 1

DragDrop 2

DragDrop 3

Views and Controls/SKView

SK Demo 1

SK Demo 2

Views and Controls/SplitView

SplitView - subview sizes

SplitView - custom1

SplitView - custom2

SplitView - custom3

20190429 FB 5.8.4

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.8.3]

FB 5.8.4 requires OS X 10.7+ for CocoaUI apps. Non-CocoaUI requires 10.6

[01] New examples added:

(a) 'List Metal-capable Devices' in Graphics directory

(b) Timer examples in Processes and Threads/Timer

(c) Several KeyLoop demos in /EventHandling/KeyViewLoop

[02] SQLiteDemoExpanded.bas demo and its companion Tlbox SQLite3Expanded.incl updated. Still in "Non-CocoaUI Demos" because it doesn't require CocoaUI.

[03] In addition to code signing, FBtoC is now notarized and uses Apple's "hardened" runtime. FBtoC has been code signed for a while now.

CocoaUI changes/additions and new demos

Changes that may break existing apps

AVKit

Header name changed from "Tlbox CocoUI_AVKit.incl" to "Tlbox AVKit.incl"

Contacts

Header name changed from "Tlbox CocoUI_Contacts.incl" to "Tlbox Contacts.incl"

ContactsUI

Header name changed from "Tlbox CocoUI_ContactsUI.incl" to "Tlbox ContactsUI.incl"

CoreLocation

Header name changed from "Tlbox CocoUI_CoreLocation.incl" to "Tlbox CoreLocation.incl"

KeyedUnarchiver

KeyedUnarchiverInitForReadingFromData function name changed to KeyedUnarchiverForReadingFromData

MapKit

Header name changed from "Tlbox CocoUI_MapKit.incl" to "Tlbox MapKit.incl"

Metal

Header name changed from "Tlbox CocoUI_Metal.incl" to "Tlbox Metal.incl"

MTKView
MTKViewInitWithFrame function name changed to MTKViewWithFrame

PDFKit
Header name changed from "Tlbx CocoUI_PDFKit.incl" to "Tlbx PDFKit.incl"

Application
additions

AttributedString
additions

Calendar
addition

CharacterSet
additions

Color
addition

Control
additions

FileManager
additions

Font
additions

Image
additions

KeyedArchiver
additions

KeyedUnarchiver
additions

change
KeyedUnarchiverInitForReadingFromData function name changed to KeyedUnarchiverForReadingFromData

Locale
additions

MTKView
change
MTKViewInitWithFrame function name changed to MTKViewWithFrame

Port
new

RunLoop
new

Sound
additions

TextContainer
additions

TextField
additions

TextStorage
additions

TextView
additions

Thread
additions

Timer
change

For convenience, a timer callback function now takes an optional *userInfo* param

TimeZone
additions

URL
additions

View
addition

Window
additions

New demos

Dates and Times
 CalendarEnumerateDates

EventHandlering/KeyViewLoop
 DoCommandBySelector
 KeyViewLoop by Tag
 KeyViewLoop ordered by tag

Views and Controls/MapView/
 MapView
 MapView - directions
 MapView (nib)

Views and Controls/WebView/
 WebView
 WebView (nib)

20190319 FB 5.8.3
<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.8.2]

FB 5.8.3 requires OS X 10.7+ for CocoaUI apps. Non-CocoaUI requires 10.6

[01] Util_String.incl conditionally includes(via the new _COCOAUI constant) NSNotFound only when not previously defined (usually by CocoaUI).
This fix and similar changes in Util_FileManager.incl, Util_PathUtilities.incl and Util_URL.incl should eliminate redefined errors when implementing a new FB release.
[02] FBtoC's "Make Xcode Project" option use a new 64-bit template for those converting to Xcode.

CocoaUI changes/additions and new demos

Changes that may break existing apps

AVKit
All function names and variable types renamed to have 'AV' prefix.

Appearance

addition

AttributedString

additions

AVKit

changes

All function names and variable types now have 'AV' prefix.

Contacts

new

ContactsUI

new

CoreLocation

new

Image

addition

MapKit

new

TableView

fix

Editable cells in a multi-column table view didn't update table data.

URLRequest

new

URLSession

new

View

change

ViewEffectiveAppearance function can now take zero tag value. When zero is specified, an off-screen, autoreleased view will be used.

WebKit

new

New demos

Networking/

URLSession

FB 5.8.2 requires OS X 10.7+ for CocoaUI apps. Non-CocoaUI require 10.6

[01] Build Settings' Compiler Options popup menu defaults to '-m64 -DDECARBONATE' instead of 'None' and 10.14(if available) BaseSDK.

Primary goal: Decrease newbie's steps to run an initial simple app(like "Hello World") by avoiding 32-bit options and the folderol entailed such as the 10.6 SDK quest and FixSDK's subsequent voodoo.

[02] FB now supports Objective-C's literal syntax (i.e. without resorting to BeginC/EndC). Numbers, Arrays and Dictionaries are all supported. 'Literal Syntax Demos' is available in CocoaUI Demos

[03] NSLog.incl updated to display white text in Mojave's dark mode.

[04] A 'IsDarkMode' example is now available (CocoaUI Demos - miscellaneous directory).

[05] FinderInfo keyword translation fixed.

20190212 FB 5.8.1

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.8]

FB 5.8.1 requires OS X 10.7+ for CocoaUI apps. Non-CocoaUI require 10.6

[01] Crash using FBtoC's "Make Xcode Project" caused by index out of bounds. Fix was applied to 5.8 binary shortly after its release. However, this doesn't fix the overall broken state of the "Make Xcode Project" (with stripping)

[02] Apple Carbon Help calls such as AHGotoPage() were failing to link during a build. Carbon framework was omitted from link commands but will have to be removed when macOS 10.15 arrives (because Carbon will be gone). Fix was applied to 5.8 binary shortly after its release.

[03] New version (1.30) of FixSDK fixes two issues:
A. Clicking Fix All or Restore All would attempt to fix/restore a 10.14 sdk.
B. Fix All or Restore All buttons would sometimes be active even when all sdks were fixed or restored.
FB 5.8.0 was updated with the new FixSDK and supplied to the list on 5-Feb-2019.

[04] Couple more characters added to FBtoC's filename character whitelist. Added apostrophe.

[05] Bug: 'Edit Menu 2' translated incorrectly. Fixed.

[06] Internal buglet: Function prototypes and code with no parameters now generates '(void)' instead of '()'.
Also some static cleanup of same issue for files within build_goodies/CocoaUI.

[07] TWM, after an illustrious career building Carbon user interfaces, is retired and no longer available in FB.

Those still building Carbon user interfaces should save their own private copy.

N.B. TWM is obsolete and its author no longer supports it.

CocoaUI changes/additions and new demos

--- Changes that may break existing apps ---

StringFromRange function name changed to **RangeToString**

---- New stuff/changes ----

Array

addition

Bundle

additions

CharacterSet

additions

ComparisonPredicate

new

CompoundPredicate

new

DataDetector

new

Expression

new

Graphics
additions

KeyedArchiver
additions

KeyedUnarchiver
additions

LinguisticTagger
addition

NibMenu
NibPopover
NibToolbar
NibView
NibWindow

change:
The *nibName* param in the above statements can now optionally take a file extension.
If a file extension is not specified, either *'xib'* or *'nib'* is used.

Range
new

Text
additions

UserDefaults
additions

View
additions

---- New demos ---
Collections/Array
ShuffledArray

Filters and Sorting/Expression
Expression Evaluator

Strings and Text/HTML
TextView HTML

Strings and Text/RegularExpression/
Regex Find Link

20190127 FB 5.8.0

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.120]

FB 5.8.0 requires OS X 10.7+ for CocoaUI apps. Non-CocoaUI require 10.6

FB's version bump to 5.8.0 reflects a gradual move to exclusively 64-bit builds. It also includes early steps to update various aspects of internal code.

[01] FBtoC now accepts and builds CFString constants and DATA statements longer than 255 characters. Key internal data structures and dependent code were switched away from length-limited pascal strings to achieve this.

[02] 32-bit builds are now deprecated and now generate a warning to switch to 64-bit builds. Please note: FBtoC/FB will build 64-bit apps exclusively at some future date and will **not** generate 32-bit apps.
Users with 32-bit apps have a couple of choices:
a. Save an FB version that supports 32-bit builds

b. Update your code to support 64-bit builds.

[03] FBtoC now accepts both .xib and .nib files.

[04] Crash using FBtoC's "Make Xcode Project..." caused by early object release is fixed.

[05] FBtoC's unix utility, mkdir(), fails if the path name contains any unacceptable characters; this failure can generate obscure FBtoC crashes. Code now rejects with an error message any source path containing characters not in:
A-Z(upper/lower case) digits 0-9, underscore, hyphen, asterisk, percent symbol, hash symbol, '\$' sign, close/open parenthesis, backslash, forward slash and space. Since I haven't found a list of characters mkdir() accepts, it's possible this character list will need amending. From my experience one unacceptable is 'f'. If any of the "acceptable" characters are found "unacceptable" by mkdir(), please post on list so FBtoC can be updated. Thank you.

[06] Redundant includes(for FBtoC.h and Runtime.h) removed from FBtoC's internal versions of General.c, Containers.c, FileHandling.c, FileHandlingUtils.h/.m and AppThings.m. Please note: these changes were tailored to FBtoC's needs and were not made to build_goodies versions of these files.

07] Several internal FBtoC bugs fixed.

20181112 FB 5.7.120

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.119]

FB 5.7.120 requires OS X 10.7+ for CocoaUI apps. Non-CocoaUI require 10.6

[01] FixSDK updated to prevent fixing or restoring a 10.14 SDK by hiding the Fix/Restore button for it.

This update was pushed out on 26-Oct-2018 and updated the FB 5.7.119 download.

CocoaUI changes/additions and new demos

Changes that may throw up warnings

TextView

TextViewTextStorage function return value changed from CFMutableAttributedStringRef to TextStorageRef

A cast may now be required to silence compiler warning

Changes that may break existing apps

FontManager

FontManagerSetSelectedFont(CTNSFontRef font) now takes an extra param FontManagerSetSelectedFont(CTNSFontRef font, Boolean isMultiple)

TextView

The textview statement now has an extra 'subclass' param. **textview** tag, rect, scrollViewTag, textContainerRef, subclass, wndTag

Workspace

WorkspaceFileTypeConformToType name changed to WorkspaceFileTypeConformsToType (typo fix)

Application

addition

Array

additions

AttributedString

additions

BitmapImageRep
additions

Color
additions

ColorList
additions and change

ColorSpace
new

Compiler
new

Dictionary
additions

DraggingInfo
additions

Error
addition

FileManager
additions

FontDescriptor
additions

FontManager
additions

change:
FontManagerSetSelectedFont now takes an extra param FontManagerSetSelectedFont(CTNSFontRef font, Boolean isMultiple)

Geometry
additions:
StringFromRange(CFRange range) = CFStringRef
RangeFromString(CFStringRef string) = CFRange

GestureRecognizer
additions

GlyphGenerator
new

Gradient
additions

Image
addition

LayoutManager
additions

Menu
additions

MenuItem
additions

PopUpButton
additions

RegularExpression
new

Scroller
additions

ScrollView
additions

SecureTextField
addition

SearchField
additions

SegmentedControl
additions

SplitView
additions

String
additions

TextCheckingResult
new

TextList
new

TextStorage
additions

TextTable
new

TextView
changes:

TextViewTextStorage now returns a TextStorageRef
The textview statement now has an extra 'subclass' param. **textview** *tag, rect, scrollViewTag, textContainerRef, subclass, wndTag*

additions

TokenField
additions

Toolbar
additions

ToolbarItem
additions

UserDefaultsController
new

View
additions

Window
additions

WindowTab
new

WindowTabGroup
new

Workspace
additions

typo fix:
WorkspaceFileTypeConformToType name corrected to WorkspaceFileTypeConformsToType

New demos

Graphics/Gradient
 Gradient3
 Gradient4

Numbers
 Value Demo

Processes and Threads/Task
 Codesign Verifier II

Strings and Text/Regular Expression
 Regex Find Years.bas
 Regex Find Numbers.bas
 Regex Email Verifier

Views and Controls/TextView
 Table Thingy

Views and Controls/View
 ShowDefinition

Windows and Panels/Window
 WindowTab AccessoryView

Files/Workspace
 Workspace Notifications

20180928 FB 5.7.119

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.118]

FB 5.7.119 requires OS X 10.7+ for CocoaUI apps. Non-CocoaUI require 10.6

[01] Building 64-bit and targeting an 'unfixed' SDK produces a warning sheet('missing Quickdraw headers') when setting FBtoC's BaseSDK. Fixed.
 Note: as of FB 5.7.117, 'fixing' SDKs is *not* necessary when building 64-bit apps. i.e. unfixed SDKs can be used.

CocoaUI changes/additions and new demos

Changes that may break existing apps

Dictionary

DictionaryWithContentsOfURL now takes an extra param: DictionaryWithContentsOfURL(CFURLRef url, ErrorRef *err) = CFDictionaryRef
DictionaryWriteToURL last param type changed: DictionaryWriteToURL(CFDictionaryRef dict, CFURLRef url, ErrorRef *err) = Boolean

FileManager

FileManagerURLForTemporaryDirectory name change to match NSFileManager method: FileManagerTemporaryDirectory

Window

WindowSetFrame now takes two additional params. WindowSetFrame(NSInteger tag, CGRect frame, Boolean display, Boolean animate)

Appearance
additions

Application
additions

AttributedString
additions

CAAnimation
new

CAKeyframeAnimation
new

CAMediaTiming
new

CAMediaTimingFunction
new

CAPropertyAnimation
new

CAValueFunction
new

Dictionary
changes:
DictionaryWithContentsOfURL now takes an extra param: DictionaryWithContentsOfURL(CFURLRef url, ErrorRef *err) = CFDictionaryRef
DictionaryWriteToURL last param type changed: DictionaryWriteToURL(CFDictionaryRef dict, CFURLRef url, ErrorRef *err) = Boolean

FileManager
change:
FileManagerURLForTemporaryDirectory name change to match NSFileManager method: FileManagerTemporaryDirectory

FileWrapper
new

JSONSerialization
new

TableRowView
new

TextAttachment
new

View
additions

Window
change:
WindowSetFrame now takes two additional params. WindowSetFrame(NSInteger tag, CGRect frame, Boolean display, Boolean animate)

Workspace
addition

New demos
Menu
 Override Edit menu

UserDefaults
 UserDefaults

Views and Controls/VisualEffectView
VisualEffectView

Windows and Panels/Window
Overlay window

20180823 FB 5.7.118

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.117]

FB 5.7.118 requires OS X 10.7+ for CocoaUI apps. Non-CocoaUI require 10.6

[01] ToolbarItemExists throws up a 'duplicate symbol' error due to a function of the same name in Util_Toolbar.incl when building 5.7.117. For now the CocoaUI function is commented out and awaiting a possible later introduction.

[02] Reorganized CocoaUI runtime caused compile errors due to missing declarations in the .h file. Fixed in an updated 5.7.117

[03] Tlbg Dispatch.incl and Tlbg clock_types.incl included in an update 5.7.117

[04] FB Help for time\$/TimeCFString and date\$/DateCFString updated. Included in an update 5.7.117

[05] Stray unwanted files (build_temp, invisibles) removed from build. Included in an update 5.7.117

[06] TWM.app replaced with one that doesn't crash on quit. Included in an update 5.7.117

[07] Some of FB's Help menu items were not showing up. Code updated to NSFileManager. Included in an update 5.7.117

CocoaUI changes/additions and new demos

Changes that may break existing apps

1. Application

AppSetTimer now returns a CFRunLoopTimerRef so calls to this function must be preceded by 'fn'

2. OperationQueue

OperationQueueAddOperationWithFunction now takes an extra 'obj' param (can be NULL)

3. Window

WindowCascadeTopLeftFromPoint now returns a CGPoint value therefore calls to this function must be preceded by 'fn'

Alert

changes:

An alert is automatically removed from the alert list once dismissed so there is now no need to call AlertRemove.

addition:

AlertWindow(NSInteger tag) = CocoaWindowRef

Application

change:

AppSetTimer now returns a CFRunLoopTimerRef

AVKit

New headers and additions to existing headers.

ColorList

new:

See ColorList.incl header and demo

ColorPanel

addition:

ColorPanelAttachColorList(ColorListRef cl)

ColorPanelDetachColorList(ColorListRef cl)

Date

addition:

DateStringWithFormat(CFStringRef format) = CFStringRef

LevelIndicator

fix:

The value of an existing levelindicator can now be changed using the levelindicator statement

MenuItem

addition:

MenuItemSetTitle(NSInteger menuIndex, NSInteger itemIndex, CFStringRef title)

Object

additions:

ObjectPerformSelector(CTypeRef obj, CFStringRef selector, CTypeRef withObj, CTimeInterval afterDelay)

ObjectCallFunction(ptr fnAddress, CTypeRef withObj, CTimeInterval afterDelay)

OperationQueue

change:

OperationQueueAddOperationWithFunction now takes a third param (obj)

Pasteboard

additions:

PasteboardWithName(CFStringRef name) = CocoaPasteboardRef

PasteboardWithUniqueName = CocoaPasteboardRef

PasteboardReleaseGlobally(CocoaPasteboardRef pb)

PopUpButton

change:

When an array or semicolon delimited list of strings is used in the popupbutton statement’s items parameter, a dash character (-) is now converted to a menu separator.

ProgressIndicator

fix:

The value of an existing progressindicator can now be changed using the progressindicator statement

RulerMarker

new:

See RulerMarker.incl header

RulerView

new:

See RulerView.incl header and demo

ScrollView

additions:

ScrollViewHorizontalRulerView(NSInteger tag) = RulerViewRef

ScrollViewSetHorizontalRulerView(NSInteger tag, RulerViewRef ruler)

ScrollViewVerticalRulerView(NSInteger tag) = RulerViewRef

ScrollViewSetVerticalRulerView(NSInteger tag, RulerViewRef ruler)

Slider

fix:

The value of an existing slider can now be changed using the slider statement

Stepper

fix:

The value of an existing stepper can now be changed using the stepper statement

TextView

additions:

TextViewDefaultParagraphStyle(NSInteger tag) = ParagraphStyleRef

TextViewSetDefaultParagraphStyle(NSInteger tag, ParagraphStyleRef style)

Window

change:

WindowCascadeTopLeftFromPoint now matches the NSWindow method and returns a CGPoint value

additions:

WindowIsOpaque(NSInteger tag) = Boolean

WindowSetOpaque(NSInteger tag, Boolean flag)

WindowEndSheetAlert(NSInteger tag, NSInteger alertTag)// macOS 10.9+

New demos

Views and Controls/TextView/

RulerViewRegisterUnit

TextView Selection

Windows and Panels/ColorPanel/

ColorListDemo

20180724 FB 5.7.117

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.116]

FB 5.7.117 requires OS X 10.7+ for CocoaUI apps. Non-CocoaUI require 10.6

[01] FBtoC's "Make Xcode Project..." would warn to add the Quartz.framework manually to the generated Xcode project. FBtoC now asks clang to link against the Quartz.framework which solves the problem and eliminates the warning. The 'include library "Quartz"' line of code was removed from Rntm Appearance.incl since the FBtoC change renders it unnecessary. This fix applies to standard FB user app builds too.

[02] The multi-dimensional object array (MDAArray.incl) utilities were updated to support multiple indices. See FB Examples (FB_5_7_117_Examples/Non-CocoaUI demos/CoreFoundation/MDAArray (multi-dimensional)) for the include and demos.

[03] FB Help Additions and Changes

- (a) FB's new 'for..in' object iterator loop now documented in the 'for' loop section
- (b) FB's dialog function and user-designated event handling function (i.e. fn DoDialog) are detailed more in the dialog(CocoaUI) section.
- (c) The 'on dialog' page was updated.

[04] Renamed header, TextServices.tlbg, to Tlbg TextServices.incl

[05] IOKit Headers (OSX IOKitLib.incl, OSX IOSerialKeys.incl, OSX termios.incl) updated

[06] 'AccessoryView with CocoaUI' example (Examples/Non-CocoaUI demos/Files/OSPanel examples) fails in 64-bit updated to use cocoainit.

[07] New NSLog example shows debugging macro usage. Currently in: Non-CocoaUI demos/Debug/Debugging Macros and /CocoaUI demos/Debug

CocoaUI changes/additions and new demos

Changes that may break your app

1. AffineTransform

AffineTransform function name changed to AffineTransformInit

2. BezierPath

BezierPath function name changed to BezierPathInit

3. IndexSet

IndexSet function name changed to IndexSetInit

4. PrintPanel

PrintPanel function name changed to PrintPanelInit

5. ProcessInfo

ProcessInfo function name changed to ProcessInfoInit

6. TextContainer

TextContainerInit function name changed to TextContainerWithSize

7. TextView

The **textview** statement change - now takes an extra param
textview tag, rect, scrollTag, textContainerRef, wndTag

AffineTransform

function name change:
AffineTransformInit - was AffineTransform

AttributedString

addition:
AttributedStringAppendCharacters(CFMutableAttributedStringRef aString, CFStringRef string)

BezierPath

function name change:
BezierPathInit - was BezierPath

BitmapImageRep

new:
See BitmapImageRep.incl header

CIDetector

new:
See CIDetector.incl header and demo

CIFilter

new:
See CIFilter.incl header and demo

CImage

new:
See CImage.incl header and demo

CIQRCodeFeature

new:
See CIQRCodeFeature.incl header and demo

ImageRep

new:
See ImageRep.incl header

ImageView

dialog event:
The imageview now triggers _btnClick dialog event when an image file is dropped onto an editable view

IndexSet

function name change:
IndexSetInit - was IndexSet

LayoutManager

new:
See LayoutManager.incl header and demo

Menu

param change:
MenuSetTag(NSInteger menuIndex, NSInteger tag) - itemIndex param removed

additions:
MenuBarVisible = Boolean
MenuIsHidden(NSInteger menuIndex) = Boolean
MenuSetHidden(NSInteger menuIndex, Boolean flag)
MenuIsHiddenOrHasHiddenAncestor(NSInteger menuIndex) = Boolean

MenuItem

additions:
MenuItemIsHidden(NSInteger menuIndex, NSInteger itemIndex) = Boolean
MenuItemSetHidden(NSInteger menuIndex, NSInteger itemIndex, Boolean flag)
MenuItemIsHiddenOrHasHiddenAncestor(NSInteger menuIndex, NSInteger itemIndex) = Boolean

Number

addition:

NumberObjCType(CFNumberRef number) = CFStringRef

OpenPanel

new:
See OpenPanel.incl header and demos

PDFInfo

new:
See PDFInfo.incl header

PDFPanel

new:
See PDFPanel.incl header

PrintPanel

function name change:
PrintPanellnit - was PrintPanel

ProcessInfo

function name change:
ProcessInfolnit - was ProcessInfo

SavePanel

new:
See SavePanel.incl header and demos

String

additions:
StringWithCString(const char *cString, NSStringEncoding encoding) = CFStringRef
StringWithUTF8String(const char *cString) = CFStringRef
StringCStringUsingEncoding(CFStringRef string, NSStringEncoding enc) = ptr

Text

additions:
TextMaxSize(NSInteger tag) = CGSize
TextSetMaxSize(NSInteger tag, CGSize size)
TextMinSize(NSInteger tag) = CGSize
TextSetMinSize(NSInteger tag, CGSize size)
TextIsVerticallyResizable(NSInteger tag) = Boolean
TextSetVerticallyResizable(NSInteger tag, Boolean flag)
TextIsHorizontallyResizable(NSInteger tag) = Boolean
TextSetHorizontallyResizable(NSInteger tag, Boolean flag)
TextSizeToFit(NSInteger tag)

TextContainer

change:
TextContainerWithSize(CGSize size) = TextContainerRef - was TextContainerInit

TextStorage

new:
See TextStorage.incl header and demo

TextView

addition:
TextViewSetWordWrap(NSInteger tag, Boolean flag)

statement change:
textview tag, rect, scrollTag, textContainerTag, wndTag

Timer

new:
See Timer.incl header and demos

View

addition:
ViewExists(NSInteger tag) = Boolean
ViewVisibleRect(NSInteger tag) = CGRect

New demos

Graphics/Core image
QR Coder
QR Code Reader

Graphics/Images and PDF
BitmapImageRep Demo

Task Management/Timer
TimerScheduledWithInterval
TimerWithFireDate

Views and Controls/TextView
TextView - Wordwrap
TextView - Two Columns

20180618 FB 5.7.116

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.115]

FB 5.7.116 requires OS X 10.7+ for CocoaUI apps. Non-CocoaUI require 10.6

[01]Fast Enumeration 'For...in' Loop
Collection types currently supported: CFArray, CFMutableArray, CFDictionary,
CFMutableDictionary, CFSet, CFMutableSet, OrderedSet, MutableOrderedSet, CountedSet,
PointerArray.
[02] Stop statement changed (AppThings.c) to use kCFStringEncodingUTF8 for message string
[03] Bug fix to OPEN "I" to allow 'file not found' reporting with FB's file I/O error handling.

CocoaUI changes/additions and new demos

Changes that may break your app

1. BundleExecutableURL name changed due to typo.
2. GraphicsShowAnimationEffect name changed to NSGraphicsShowAnimationEffect

Alert

addition:

AlertAddButtonWithTitle(NSInteger tag, CFStringRef title)

AppleEventDescriptor

new:

See AppleEventDescriptor.incl header

Array

additions:

ArrayFirstObjectCommonWithArray(CFArrayRef array1, CFArrayRef array2) = CFTypeRef

ArraysIsEqualToArray(CFArrayRef array1, CFArrayRef array2) = Boolean

ArrayByAddingObject(CFArrayRef array, CFTypeRef obj) = CFArrayRef

ArrayByAddingObjectsFromArray(CFArrayRef array1, CFArrayRef array2) = CFArrayRef

ArraySubarrayWithRange(CFArrayRef array, CFRange range) = CFArrayRef

ArraySortedArrayUsingSelector(CFArrayRef array, CFStringRef selector) = CFArrayRef

Bundle

typo fix:

BundleExecutableURL

additions:

BundleURLsForResources(BundleRef bundle, CFStringRef extension, CFStringRef subdirectory) = CFArrayRef

BundleImageForResource(BundleRef bundle, CFStringRef name) = ImageRef

BundleURLForSoundResource(BundleRef bundle, CFStringRef name) = CFURLRef

BundleLocalizedString(BundleRef bundle, CFStringRef key, CFStringRef value, CFStringRef table) = CFStringRef

BundleContextHelp(BundleRef bundle, CFStringRef key) = CFAttributedStringRef
BundleExecutableURL(BundleRef bundle) = CFURLRef
BundlePrivateFrameworksURL(BundleRef bundle) = CFURLRef
BundleSharedFrameworksURL(BundleRef bundle) = CFURLRef
BundleBuiltInPlugInsURL(BundleRef bundle) = CFURLRef
BundleURLForAuxiliaryExecutable(BundleRef bundle, CFStringRef name) = CFURLRef
BundleSharedSupportURL(BundleRef bundle) = CFURLRef
BundleAppStoreReceiptURL(BundleRef bundle) = CFURLRef
BundleLocalizations(BundleRef bundle) = CFArrayRef
BundlePreferredLocalizations(BundleRef bundle) = CFArrayRef
BundleDevelopmentLocalization(BundleRef bundle) = CFStringRef
BundleLocalizedInfoDictionary(BundleRef bundle) = CFDictionaryRef
BundlePreferredLocalizationsFromArray(CFArrayRef localizationsArray) = CFArrayRef
BundlePreferredLocalizationsFromArrayForPreferences(CFArrayRef localizationsArray, CFArrayRef preferencesArray) = CFArrayRef

ByteCountFormatter

new:
See ByteCountFormatter.incl header and demo

CAGradientLayer

new:
See CAGradientLayer.incl header and demo

CALayer

new:
See CALayer.incl header and demos

Calendar

new:
See Calendar.incl header

CAReplicatorLayer

new:
See CAReplicatorLayer.incl header and demos

CAScrollLayer

new:
See CAScrollLayer.incl header and demo

CAShapeLayer

new:
See CAShapeLayer.incl header and demo

CATextLayer

new:
See CATextLayer.incl header and demo

CATransform3D

new:
See CATransform3D.incl header

CATransformLayer

new:
See CATransformLayer.incl header and demos

ClickGestureRecognizer

new:
See ClickGestureRecognizer.incl header and demos

ColorPanel

new:
_colorPanelChangeColor dialog event

additions:
ColorPanellsContinuous = Boolean
ColorPanelSetContinuous(Boolean flag)

ColorPanelMode = NSColorPanelMode
ColorPanelSetActionCallback(ptr callback, ptr userData)
ColorPanelShowsAlpha = Boolean

CountedSet

new:
See CountedSet.incl header and demo

Calendar

new:
See Calendar.incl header

ColorPanel

additions:
ColorPanellsContinuous = Boolean
ColorPanelSetContinuous(Boolean flag)
ColorPanelMode = NSColorPanelMode
ColorPanelSetActionCallback(ptr callback, ptr userData)
ColorPanelShowsAlpha = Boolean

ComboBox

additions:
ComboBoxHasVerticalScroller(NSInteger tag) = Boolean
ComboBoxIntercellSpacing(NSInteger tag) = CGSize
ComboBoxIsBordered(NSInteger tag) = Boolean
ComboBoxItemHeight(NSInteger tag) = CGFloat
ComboBoxNumberOfVisibleItems(NSInteger tag) = NSInteger
ComboBoxDeselectItemAtIndex(NSInteger tag, NSInteger index)

Control

addition:
ControlLineBreakMode(NSInteger tag) = NSLineBreakMode

Data

new:
See Data.incl header

Date

additions:
DateDistantFuture = CFDateRef
DateDistantPast = CFDateRef
DateIsEqualToDate(CFDateRef dt1, CFDateRef dt2) = Boolean
DateEarlierDate(CFDateRef dt1, CFDateRef dt2) = CFDateRef
DateLaterDate(CFDateRef dt1, CFDateRef dt2) = CFDateRef
DateCompare(CFDateRef dt1, CFDateRef dt2) = NSComparisonResult
DateTimeIntervalSinceDate(CFDateRef dt1, CFDateRef dt2) = CFTimeInterval
DateTimeIntervalSinceNow(CFDateRef dt) = CFTimeInterval
DateTimeIntervalSinceReferenceDate(CFDateRef dt) = CFTimeInterval
DateTimeIntervalSince1970(CFDateRef dt) = CFTimeInterval
DateByAddingTimeInterval(CFDateRef dt, CFTimeInterval secs) = CFDateRef
DateDescription(CFDateRef dt) = CFStringRef
DateDescriptionWithLocale(CFDateRef dt, CFLocaleRef locale) = CFStringRef

DateComponents

new:
See DateComponents.incl header and demo

DateComponentsFormatter

new:
See DateComponentsFormatter.incl header

DateFormatter

new:
See DateFormatter.incl header and demo

DateInterval

new:

See DateInterval.incl header

DateIntervalFormatter

new:
See DateIntervalFormatter.incl header

DatePicker

additions:
DatePickerCalendar(NSInteger tag) = CFCalendarRef
DatePickerSetCalendar(NSInteger tag, CFCalendarRef cal)

DecimalNumber

new:
See DecimalNumber.incl header and demos

DialogEvent

addition:
DialogEventColor = ColorRef

Dictionary

additions:
DictionaryKeysSortedByValueUsingSelector(CFDictionaryRef dict, CFStringRef selector) = CFArrayRef
DictionaryFileSize(CFDictionaryRef dict) = UInt64
DictionaryFileType(CFDictionaryRef dict) = CFStringRef
DictionaryFileCreationDate(CFDictionaryRef dict) = CFDateRef
DictionaryFileModificationDate(CFDictionaryRef dict) = CFDateRef
DictionaryFilePosixPermissions(CFDictionaryRef dict) = NSUInteger
DictionaryFileOwnerAccountID(CFDictionaryRef dict) = CFNumberRef
DictionaryFileOwnerAccountName(CFDictionaryRef dict) = CFStringRef
DictionaryFileGroupOwnerAccountID(CFDictionaryRef dict) = CFNumberRef
DictionaryFileGroupOwnerAccountName(CFDictionaryRef dict) = CFStringRef
DictionaryFileExtensionHidden(CFDictionaryRef dict) = Boolean
DictionaryFilesImmutable(CFDictionaryRef dict) = Boolean
DictionaryFilesAppendOnly(CFDictionaryRef dict) = Boolean
DictionaryFileSystemFileNumber(CFDictionaryRef dict) = NSUInteger
DictionaryFileHFSTypeCode(CFDictionaryRef dict) = OSType
DictionaryFileHFSCreatorCode(CFDictionaryRef dict) = OSType

FileHandle

new:
See FileHandle.incl header and demo

Font

additions:
FontUserFontOfSize(CGFloat size) = FontRef
FontBoldSystemFontOfSize(CGFloat size) = FontRef
FontControlContentFontOfSize(CGFloat size) = FontRef
FontLabelFontOfSize(CGFloat size) = FontRef
FontMenuFontOfSize(CGFloat size) = FontRef
FontMenuBarFontOfSize(CGFloat size) = FontRef
FontMessageFontOfSize(CGFloat size) = FontRef
FontMonospacedDigitSystemFontOfSize(CGFloat size, CGFloat weight) = FontRef
FontPaletteFontOfSize(CGFloat size) = FontRef
FontSystemFontOfSize(CGFloat size) = FontRef
FontSystemFontOfSizeAndWeight(CGFloat size, CGFloat weight) = FontRef
FontTitleBarFontOfSize(CGFloat size) = FontRef
FontToolTipsFontOfSize(CGFloat size) = FontRef
FontDescriptor(FontRef font) = FontDescriptorRef
FontLabelFontSize = CGFloat
FontSmallSystemFontSize = CGFloat
FontSystemFontSize = CGFloat
FontSystemFontSizeForControlSize(NSControlSize size) = CGFloat
FontAscender(FontRef font) = CGFloat
FontBoundingRectForFont(FontRef font) = CGRect
FontCapHeight(FontRef font) = CGFloat
FontDescender(FontRef font) = CGFloat
FontItalicAngle(FontRef font) = CGFloat

FontLeading(FontRef font) = CGFloat
FontMatrix(FontRef font) = ptr
FontSize(FontRef font) = CGFloat
FontUnderlinePosition(FontRef font) = CGFloat
FontUnderlineThickness(FontRef font) = CGFloat
FontXHeight(FontRef font) = CGFloat
FontDisplayName(FontRef font) = CFStringRef
FontFamilyName(FontRef font) = CFStringRef
FontName(FontRef font) = CFStringRef
FontSetUserFont(FontRef font)

FontManager

additions:
FontManagerAvailableFonts = CFArrayRef
FontManagerAvailableFontFamilies = CFArrayRef
FontManagerAvailableFontNamesWithTraits(NSFontTraitMask traits) = CFArrayRef
FontManagerAvailableMembersOfFontFamily(CFStringRef fam) = CFArrayRef

GestureRecognizer

new:
See GestureRecognizer.incl header and demos

Graphics

addtions:
NSDrawThreePartImage(CGRect frame, ImageRef startCap, ImageRef centerFill, ImageRef endCap, Boolean vertical, NSCompositingOperation op, CGFloat alphaFraction, Boolean flipped)
NSDrawNinePartImage(CGRect frame, ImageRef tl, ImageRef t, ImageRef tr, ImageRef l, ImageRef c, ImageRef r, ImageRef bl, ImageRef b, ImageRef br, NSCompositingOperation op, CGFloat alphaFraction, Boolean flipped)
NSEraseRect(CGRect r)
NSFrameRect(CGRect r)
NSFrameRectWithWidth(CGRect r, CGFloat frameWidth)
NSFrameRectWithWidthUsingOperation(CGRect r, CGFloat frameWidth, NSCompositingOperation operation)
NSRectFillUsingOperation(CGRect r, NSCompositingOperation operation)
NSRectFill(CGRect r)

change:
GraphicsShowAnimationEffect function name changed to NSShowAnimationEffect

IndexPath

new:
See IndexPath.incl header

IndexSet

additions:
IndexSetEnumerateRangesInRange(IndexSetRef set, CFRange range, NSEnumerationOptions options, ptr callback, ptr userData)
IndexSetEnumerateRanges(IndexSetRef set, ptr callback, ptr userData)
IndexSetEnumerateRangesWithOptions(IndexSetRef set, NSEnumerationOptions options, ptr callback, ptr userData)
IndexSetIndexLessThan(IndexSetRef set, NSUInteger value) = NSUInteger
IndexSetIndexLessThanOrEqualTo(IndexSetRef set, NSUInteger value) = NSUInteger
IndexSetIndexGreaterThanThanOrEqualTo(IndexSetRef set, NSUInteger value) = NSUInteger
IndexSetIndexGreaterThan(IndexSetRef set, NSUInteger value) = NSUInteger
IndexSetEnumerateIndexes(IndexSetRef set, ptr callback, ptr userData)
IndexSetEnumerateIndexesWithOptions(IndexSetRef set, NSEnumerationOptions options, ptr callback, ptr userData)
IndexSetEnumerateIndexesInRange(IndexSetRef set, CFRange range, NSEnumerationOptions options, ptr callback, ptr userData)

LinguisticTagger

new:
See LinguisticTagger.incl and demo

Locale

new:
See Locale.incl header

MagnificationGestureRecognizer

new:

See MagnificationGR.incl header and demos

NSDecimal

new:
See NSDecimal.incl header and demo

NSHFSFileTypes

new:
See NSHFSFileTypes.incl header

OrderedSet

new:
See OrderedSet.incl header

PanGestureRecognizer

new:
See PanGestureRecognizer.incl header and demos

ParagraphStyle

new:
See ParagraphStyle.incl header

Pasteboard

additons:
PasteboardRefClearContents(CocoaPasteboardRef pb) = NSInteger
PasteboardRefWriteObjects(CocoaPasteboardRef pb, CFArrayRef objects) = Boolean
PasteboardRefSetStringForType(CocoaPasteboardRef pb, CFStringRef string, CFStringRef type) = Boolean
PasteboardRefItems(CocoaPasteboardRef pb) = CFArrayRef
PasteboardRefPropertyListForType(CocoaPasteboardRef pb, CFStringRef type) = CFTypeRef
PasteboardRefStringForType(CocoaPasteboardRef pb, CFStringRef type) = CFStringRef
PasteboardRefTypes(CocoaPasteboardRef pb) = CFArrayRef
PasteboardRefName(CocoaPasteboardRef pb) = CFStringRef
PasteboardRefChangeCount(CocoaPasteboardRef pb) = NSInteger
PasteboardSetStringForType(CFStringRef string, CFStringRef type) = Boolean
PasteboardStringForType(CFStringRef type) = CFStringRef

PDFThumbnailView

new:
The PDFThumbnailView can now be built programmatically
pdfthumbnailview *tag, rect, pdfViewTag, wndTag*

PDFView

new:
The PDFView can now be built programmatically
pdfview *tag, rect, pdfDocRef, wndTag*

Pipe

new:
See Pipe.incl header and demo

PointerArray

new:
See PointerArray.incl header

PointerFunctions

new:
See PointerFunctions.incl header

PressGestureRecognizer

new:
See PressGestureRecognizer.incl header and demos

Printer

addition:
PrinterDeviceDescription(PrinterRef printer) = CFDictionaryRef

PrintOperation

addition:
PrintOperationRunModal(PrintOperationRef printOperation, NSInteger wndTag, ptr callback, ptr userData)

PrintPanel

addition:
PrintPanelBeginSheet(PrintPanelRef pp, PrintInfoRef printInfo, NSInteger wndTag, ptr callback, ptr userData)

ProcessInfo

new:
See ProcessInfo.incl header

ProgressIndicator

additions:
ProgressIndicatorUsesThreadedAnimation(NSInteger tag) = Boolean
ProgressIndicatorControlSize(NSInteger tag) = NSControlSize
ProgressIndicatorControlTint(NSInteger tag) = NSControlSize
ProgressIndicatorSetControlTint(NSInteger tag, NSControlTint tint)
ProgressIndicatorIsBezeled(NSInteger tag) = Boolean
ProgressIndicatorSetBezeled(NSInteger tag, Boolean flag)
ProgressIndicatorIsIndeterminate(NSInteger tag) = Boolean
ProgressIndicatorStyle(NSInteger tag) = NSUInteger
ProgressIndicatorSizeToFit(NSInteger tag)
ProgressIndicatorIsDisplayWhenStopped(NSInteger tag) = Boolean

PropertyListSerialization

new:
See PropertyList.incl header and demo

RotationGestureRecognizer

new:
See RotationGestureRecognizer.incl header and demos

Scanner

new:
See Scanner.incl header and demo

Scroller

new:
See Scroller.incl header

ScrollView

additions:
ScrollViewHorizontalScroller(NSInteger tag) = ScrollerRef
ScrollViewHasHorizontalScroller(NSInteger tag) = Boolean
ScrollViewVerticalScroller(NSInteger tag) = ScrollerRef
ScrollViewHasVerticalScroller(NSInteger tag) = Boolean
ScrollViewAutohidesScrollers(NSInteger tag) = Boolean

Set

new:
See Set.incl header and demos

Slider

additions:
SliderKnobThickness(NSInteger tag) = CGFloat
SliderIsVertical(NSInteger tag) = Boolean
SliderMaxValue(NSInteger tag) = double
SliderSetMaxValue(NSInteger tag, double value)
SliderMinValue(NSInteger tag) = double
SliderSetMinValue(NSInteger tag, double value)
SliderAllowsTickMarkValuesOnly(NSInteger tag) = Boolean
SliderIndexOfTickMarkAtPoint(NSInteger tag, CGPoint pt) = NSInteger
SliderNumberOfTickMarks(NSInteger tag) = NSInteger
SliderSetNumberOfTickMarks(NSInteger tag, NSInteger value)
SliderRectOfTickMarkAtIndex(NSInteger tag, NSInteger index) = CGRect
SliderTickMarkPosition(NSInteger tag) = NSTickMarkPosition
SliderTickMarkValueAtIndex(NSInteger tag, NSInteger index) = double

StatusBar

new:
See StatusBar.incl header and demo

StatusBarButton

new:
See StatusBarButton.incl header

StatusItem

new:
See StatusItem.incl header and demo

String

additions:
StringWithBytes(ptr bytes, NSUInteger length, NSStringEncoding encoding) = CFStringRef
StringWithData(CFDataRef dta, NSStringEncoding encoding) = CFStringRef

TabView

additions:
TabViewType(NSInteger tag) = NSTabViewType
TabViewControlTint(NSInteger tag) = NSControlTint
TabViewDrawsBackground(NSInteger tag) = Boolean
TabViewMinimumSize(NSInteger tag) = CGSize
TabViewControlSize(NSInteger tag) = NSControlSize
TabViewAllowsTruncatedLabels(NSInteger tag) = Boolean
TabViewSetAllowsTruncatedLabels(NSInteger tag, Boolean flag)
TabViewPosition(NSInteger tag) = NSTabPosition
TabViewSetPosition(NSInteger tag, NSTabPosition position)
TabViewBorderType(NSInteger tag) = NSTabViewBorderType
TabViewSetBorderType(NSInteger tag, NSTabViewBorderType position)

Task

new:
See Task.incl header and demo

Text

additions:
TextReplaceCharactersInRangeWithString(NSInteger tag, CFRange range, CFStringRef string)
TextScrollRangeToVisible(NSInteger tag, CFRange range)

TextContainer

new:
See TextContainer.incl header and demo

TextTab

new:
See TextTab.incl header

TextView

additions:
TextViewTextContainer(NSInteger tag) = TextContainerRef
TextViewSetTextContainer(NSInteger tag, TextContainerRef tc)
TextViewReplaceTextContainer(NSInteger tag, TextContainerRef tc)
TextViewTextContainerInset(NSInteger tag) = CGSize
TextViewSetTextContainerInset(NSInteger tag, CGSize inset)
TextViewTextContainerOrigin(NSInteger tag) = CGPoint
TextViewInvalidateTextContainerOrigin(NSInteger tag)

URL

addition:
URLIsEqual(CFURLRef url1, CFURLRef url2) = Boolean

UserNotification

new:
See UserNotification.incl header and demos

UserNotificationAction

new:
See UserNotificationAction.incl header and demos

UserNotificationCenter

new:
See UserNotificationCenter.incl header and demos

Value

addition:
ValueWithCATransform3D(CATransform3D t) = ValueRef

View

additions:
ViewLayer(NSInteger tag) = CALayerRef
ViewSetLayer(NSInteger tag, CALayerRef layer)
ViewGestureRecognizers(NSInteger tag) = CFArrayRef
ViewAddGestureRecognizer(NSInteger tag, GestureRecognizerRef ref)
ViewRemoveGestureRecognizer(NSInteger tag, GestureRecognizerRef ref)
ViewSetClickGestureRecognizerCallback(NSInteger tag, ptr callback, ptr userData)
ViewSetMagnificationGestureRecognizerCallback(NSInteger tag, ptr callback, ptr userData)
ViewSetPanGestureRecognizerCallback(NSInteger tag, ptr callback, ptr userData)
ViewSetPressGestureRecognizerCallback(NSInteger tag, ptr callback, ptr userData)
ViewSetRotationGestureRecognizerCallback(NSInteger tag, ptr callback, ptr userData)

Window

addition:
WindowScreen(NSInteger tag) = ScreenRef

Workspace

additions:
WorkspaceLaunchApplicationAtURL(CFURLRef url, NSWorkspaceLaunchOptions options, CFDictionaryRef configuration, ErrorRef *err) = Boolean
WorkspaceDesktopImageURLForScreen(ScreenRef screen) = CFURLRef
WorkspaceSetDesktopImageURLForScreen(CFURLRef url, ScreenRef screen, CFDictionaryRef options, ErrorRef *err)
WorkspaceDesktopImageOptionsForScreen(ScreenRef screen) = CFDictionaryRef

New demos

Collections/FastEnumeration ForLoop
 Array1...Array4
 CountedSet
 Dictionary
 OrderedSet
 Set1...Set3

Data Formatting
 ByteCountFormatter

Data Formatting/Locale
 NativeCurrency

Dates and Times
 DateComponents
 DateFormatter
 DateInterval

EventHandling/Gestures
 ClickGestureRecognizer1
 ClickGestureRecognizer2
 ClickGestureRecognizer3
 MagnificationGestureRecognizer
 PanGestureRecognizer
 PressGestureRecognizer

RotationGestureRecognizer

Files

PropertyListSerialization

Files/FileHandle

FileHandle Demo

Graphics/CA

CAGradientLayer

CALayer

CAReplicatorLayer

CAScrollLayer

CAShapeLayer

CATextLayer

Menu

StatusBarItem

Miscellaneous/Task

BitStatus

Notifications

UserNotificationCenter

Numbers

DecimalNumber1

DecimalNumber2

NSDecimal

Strings and Text

LinguisticTagger

Views and Controls/TextView

TextContainer

20180502 FB 5.7.115

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.114]

FB 5.7.115 requires OS X 10.7+ for CocoaUI apps. Non-CocoaUI require 10.6

[01] CocoaUI scripts palette content updated

[02] Release Notes pdf rebuilt with an export from Pages instead of TextEdit

[03] CocoaUI's StringApplyTransform requires macOS 10.12+ but was incorrectly included for any OS version. Please note: the FB programmer is still responsible for using an appropriate SDK (10.12+). This fix was also applied and announced on list for an interim 5.7.114 build.

[04] DateCFString/TimeCFString added to runtime. Similar to date\$/time\$ (now deprecated) but accepts and returns CFStrings instead of pascal strings. See FBHelp and Examples/Miscellaneous/DateCFString Demo for more details.

[05] YAST (Yet Another Scrolling TextView) example removed from 'Examples/Text/Scrolling text views' because the bundle it depends on is 32-bit and the code is obsolete 'TXN' calls.

[06] FixSDK version 1.2.7 includes miscellaneous fixes and is now code signed for Apple's Gatekeeper.

- import/export alert sheet
- missing 10.6 SDK alert
- non-recursive search of export directory
- remember export directory after exporting a single SDK

CocoaUI changes/additions

Changes that may break existing apps

1. SegmentedControlSetSegment function - imageName (CFStringRef) param changed to image (ImageRef).
2. CharacterSetXxxx function name changes (see below).

- 3. ControlSetActionCallback now has a third param (userData).
- 4. WindowSetDelegateCallback now has a third param (userData).
- 5. WorkspaceShowSearchResultsForQueryString now returns a bool, therefore calls to this function must now be preceded by **fn**.

Alert

new:
AlertButtonSetKeyEquivalent(NSInteger alertTag, NSUInteger btnIndex, CFStringRef key)
AlertButtonSetKeyEquivalentModifierMask(NSInteger alertTag, NSUInteger btnIndex, NSUInteger mask)

AppleScript

new:
AppleScriptWithContentsOfURL(CFURLRef url, CFDictionaryRef *errInfo) = AppleScriptRef
AppleScriptWithSource(CFStringRef source) = AppleScriptRef
AppleScriptIsCompiled(AppleScriptRef script) = Boolean
AppleScriptSource(AppleScriptRef script) = CFStringRef
AppleScriptRichTextSource(AppleScriptRef script) = CFAttributedStringRef
AppleScriptCompile(AppleScriptRef script, CFDictionaryRef *errInfo) = Boolean
AppleScriptExecute(AppleScriptRef script, CFDictionaryRef *errInfo) = AppleEventDescriptorRef
AppleScriptExecuteAppleEvent(AppleScriptRef script, AppleEventDescriptorRef aeDesc, CFDictionaryRef *errorInfo) = AppleEventDescriptorRef

CharacterSet

function name changes:
CharacterSetAlphanumeric**Set**
CharacterSetCapitalizedLetter**Set**
CharacterSetControl**Set**
CharacterSetDecimalDigit**Set**
CharacterSetDecomposable**Set**
CharacterSetIllegal**Set**
CharacterSetLetter**Set**
CharacterSetLowercaseLetter**Set**
CharacterSetNewline**Set**
CharacterSetNonBase**Set**
CharacterSetPunctuation**Set**
CharacterSetSymbol**Set**
CharacterSetUppercaseLetter**Set**
CharacterSetWhitespaceAndNewline**Set**
CharacterSetWhitespace**Set**

Control

param change:
ControlSetActionCallback(NSInteger tag, ptr callback, **ptr userData**)

DirectoryEnumerator

new:
DirectoryEnumeratorDirectoryAttributes(DirectoryEnumeratorRef enumerator) = CFDictionaryRef
DirectoryEnumeratorFileAttributes(DirectoryEnumeratorRef enumerator) = CFDictionaryRef
DirectoryEnumeratorLevel(DirectoryEnumeratorRef enumerator) = NSUInteger
DirectoryEnumeratorSkipDescendants(DirectoryEnumeratorRef enumerator)

Enumerator

new:
EnumeratorAllObjects(EnumeratorRef enumerator) = CFArrayRef
EnumeratorNextObject(EnumeratorRef enumerator) = CTypeRef

Geometry

new:
StringFromPoint(CGPoint pt) = CFStringRef
StringFromSize(CGSize size) = CFStringRef
StringFromRect(CGRect rect) = CFStringRef
PointFromString(CFStringRef string) = CGPoint
SizeFromString(CFStringRef string) = CGSize
RectFromString(CFStringRef string) = CGRect

KeyedArchiver

new:

KeyedArchiverArchivedDataWithRootObject(CTypeRef obj) = CFDataRef
KeyedArchiverArchiveRootObjectToURL(CTypeRef obj, CFURLRef url) = Boolean

KeyedUnarchiver

new:
KeyedUnarchiverUnarchiveObjectWithData(CFDataRef dta) = CTypeRef
KeyedUnarchiverUnarchiveObjectWithURL(CFURLRef url) = CTypeRef

MenuItem

new:
MenuItemTag(NSInteger menuIndex, NSInteger itemIndex) = NSInteger
MenuItemSetTag(NSInteger menuIndex, NSInteger itemIndex, NSInteger tag)
MenuItemSetActionCallback(NSInteger menuIndex, NSInteger itemIndex, ptr callback, ptr userData)
MenuItemImage(NSInteger menuIndex, NSInteger itemIndex) = ImageRef
MenuItemSetImage(NSInteger menuIndex, NSInteger itemIndex, ImageRef image)
MenuItemOnStatImage(NSInteger menuIndex, NSInteger itemIndex) = ImageRef
MenuItemSetOnStatImage(NSInteger menuIndex, NSInteger itemIndex, ImageRef image)
MenuItemOffStatImage(NSInteger menuIndex, NSInteger itemIndex) = ImageRef
MenuItemSetOffStatImage(NSInteger menuIndex, NSInteger itemIndex, ImageRef image)
MenuItemMixedStatImage(NSInteger menuIndex, NSInteger itemIndex) = ImageRef
MenuItemSetMixedStatImage(NSInteger menuIndex, NSInteger itemIndex, ImageRef image)
MenuItemIndentationLevel(NSInteger menuIndex, NSInteger itemIndex) = NSInteger
MenuItemSetIndentationLevel(NSInteger menuIndex, NSInteger itemIndex, NSInteger level)

NumberFormatter

new:
See NumberFormatter.incl header

PDFView

new:
See PDFKit headers and PDFView demos

PopUpButton

new:
PopUpButtonSelectItemWithTag(NSInteger popUpTag, NSInteger itemTag)
PopUpButtonIndexOfItemWithTag(NSInteger popUpTag, NSInteger itemTag) = NSInteger
PopUpButtonIndexOfItemWithTitle(NSInteger tag, CFStringRef title) = NSInteger

ScrollView

new:
ScrollViewFindBarPosition(NSInteger tag) = NSScrollViewFindBarPosition
ScrollViewSetFindBarPosition(NSInteger tag, NSScrollViewFindBarPosition position)

SegmentedControl

function param change:
SegmentedControlSetSegment(NSInteger tag, NSInteger index, NSInteger enabled, CFStringRef label, **ImageRef image**,
NSUInteger imageScaling, NSInteger width)

String

new:
StringComponentsSeparatedByCharactersInSet(CFStringRef string, CFCharacterSetRef set) = CFArrayRef
StringByTrimmingCharactersInSet(CFStringRef string, CFCharacterSetRef set) = CFStringRef
StringRangeOfString(CFStringRef string, CFStringRef searchString) = CFRange

Text

new:
TextSetAlignment(NSInteger tag, NSTextAlignment alignment)
TextSetFontWithName(NSInteger tag, CFStringRef name, CGFloat size)
TextSetFontWithNameInRange(NSInteger tag, CFStringRef name, CGFloat size, CFRange range)

TextView

new:
TextViewOrderFrontLinkPanel(NSInteger tag)
TextViewOrderFrontListPanel(NSInteger tag)
TextViewOrderFrontSpacingPanel(NSInteger tag)
TextViewOrderFrontTablePanel(NSInteger tag)
TextViewOrderFrontSubstitutionsPanel(NSInteger tag)

URL - see demos

new:

URLByResolvingBookmarkData(CFDataRef bookmarkData, NSURLBookmarkResolutionOptions options, CFURLRef relativeURL, Boolean *isStale, ErrorRef *err) = CFURLRef
URLBookmarkDataWithContentsOfURL(CFURLRef url, ErrorRef *err) = CFDataRef
URLBookmarkDataWithOptions(CFURLRef url, NSURLBookmarkCreationOptions options, CFArrayRef resourceKeys, CFURLRef relativeURL, ErrorRef *err) = CFDataRef
URLResourceValuesForKeysFromBookmarkData(CFArrayRef resourceKeys, CFDataRef bookmarkData) = CFDictionaryRef
URLWriteBookmarkData(CFDataRef bookmarkData, CFURLRef toURL, NSURLBookmarkFileCreationOptions options, ErrorRef *err) = Boolean

UserDefaults

new:

UserDefaultsColor(CFStringRef key) = ColorRef
UserDefaultsSetColor(CFStringRef key, ColorRef col)
UserDefaultsFont(CFStringRef key) = FontRef
UserDefaultsSetFont(CFStringRef key, FontRef font)

Window

param change:

WindowSetDelegateCallback(NSInteger tag, ptr callback, **ptr userData**)

Workspace

change:

WorkspaceShowSearchResultsForQueryString(CFStringRef string) = **Boolean**

- New demos -

Scripting/AppleScript/
 AppleScriptWithSource
 AppleScriptWithContentsOfURL

Application/AppShouldTerminate/
 AppShouldTerminate1
 AppShouldTerminate2

Collections/Array/
 ObjectEnumerator

Collections/Dictionary/
 KeyEnumerator
 ObjectEnumerator

Collections/
 SortDescriptors1
 SortDescriptors2

Files/FileManager/
 DirectoryEnumerator

UserDefaults/
 Color

Views and Controls/PDFView/
 PDFView1
 PDFView2
 PDFView3

Views and Controls/Field/
 SearchField w menu

Views and Controls/TextView/
 TextView w findbar

[ttp://www.4toc.com/fb/index.htm](http://www.4toc.com/fb/index.htm) [you will need to reload the page if your browser cached the page mentioning FB 5.7.113]

FB 5.7.114 requires OS X 10.7+ for CocoaUI apps. Non-CocoaUI require 10.6

[01] Building a single source file with compile options -m64 and -DDECARBONATE failed with a NSFileHandle clang error in the FBInforForOpenFile struct inside FBtoC.h. Changing the struct field declartion to void* (which makes it consistent with OSX SerialIO.incl) solves it.

[02] TWM was crashing on quit. Hopefully fixed.

[03] Similar fix for RECORD, LOF statements as described in FB 5.7.113's item #2 release notes (below)

[04] PrintFloat(), PrintTab(), PrintLong(), PrintSpc() runtime functions were excluded for 64-bit builds. Now included for both 32 and 64-bit builds

WARNING/N.B

WARNING/N.B.: This change supports PRINT to file and PRINT to buffer for 64-bit builds. PRINT to screen and PRINT to printer are **not** supported in 64-bit builds.

CocoaUI changes/additions

Changes that may break existing apps

1. Function name changes due to typos: BezierPathStrokeFillOvallnRect, BezierPathStrokeOvallnRect and BezierPathFillOvallnRect
2. Function name change: ButtonSetSound(tag, sndName) is now named ButtonSetSound**Named**(tag, sndName)
3. Function name change: PopUpButtonSetMenu(tag, menuIndex) is now named PopUpButtonSetMenu**Index**(tag, menuIndex)
4. Function name change due to typo: ScrollViewSetHorizontalLineScroll

User on dialog function

The user on dialog function can now optionally receive the event type, id and wndNum values as parameters instead of calling dialog(0), dialog(ev) and dialog(-1). This applies to both Carbon and CocoaUI apps.

This:

```
local fn DoDialog
    dim as long ev, id, wnd

    ev = dialog(0)
    id = dialog(ev)
    wnd = dialog(-1)
    // .....
end fn
```

can now be:

```
local fn DoDialog( ev as long, id as long, wnd as long )
    //.....
end fn
```

This on dialog change is optional. Your existing apps will continue to function as they did before.

User on menu function

The user on menu function can now optionally receive the menuID and itemID values as parameters instead of calling menu(_menuID) and menu(_itemID). This applies to both Carbon and CocoaUI apps.

This:

```
local fn DoMenu
    dim as long menuID, itemID

    menuID = menu(_menuID)
    itemID = menu(_itemID)
    // .....
end fn
```

can now be:

```
local fn DoMenu( menuID as long, itemID as long )
```

```
        //.....
    end fn
```

This on menu change is optional. Your existing apps will continue to function as they did before.

User on appevent function

The user on appevent function can now optionally receive the app event type as a parameter instead of calling AppEvent(0).

This:

```
local fn DoAppEvent
    dim as long ev

    ev = AppEvent(0)
    //.....
end fn
```

can now be:

```
local fn DoAppEvent( ev as long )
    //.....
end fn
```

This on appevent change is optional. Existing apps will continue to function as they did before.

AffineTransform

```
new:
AffineTransformStruct( AffineTransformRef tx ) = NSAffineTransformStruct
```

Alert

```
new:
AlertInformativeText( NSInteger tag ) = CFStringRef
AlertSetInformativeText( NSInteger tag, CFStringRef string )
AlertMessageText( NSInteger tag ) = CFStringRef
AlertSetMessageText( NSInteger tag, CFStringRef string )
```

Application

```
new:
AppTerminate
AppCurrentEvent = CocoaEventRef
AppIsRunning = Boolean
AppRun
AppStop
AppSendEvent( CocoaEventRef evnt )
AppPostEvent( CocoaEventRef evnt, Boolean atStart )
AppRunModal( NSInteger wndTag ) = NSInteger
AppStopModal
AppStopModalWithCode( NSInteger returnCode )
AppAbortModal
AppRunPageLayout
AppUserInterfaceLayoutDirection = NSUserInterfaceLayoutDirection
AppWindowWithWindowNumber( NSInteger number ) = CocoaWindowRef
AppWindows = CFArrayRef
AppMainMenu = CocoaMenuRef
AppWindowsMenu = CocoaMenuRef
AppDockTile = DockTileRef
AppHelpMenu = CocoaMenuRef
AppMiniaturizeAll
AppArrangeInFront

-- convenience functions --
AppDockTileContentView = ViewRef
AppDockTileSize = CGSize
AppDockTileShowsApplicationBadge = Boolean
AppDockTileSetShowsApplicationBadge( Boolean flag )
AppDockTileBadgeLabel = CFStringRef
```

AppDockTileSetBadgeLabel(CFStringRef label)
AppDockTileDisplay

BezierPath

typo fixes:
BezierPathStrokeFillOvalInRect
BezierPathStrokeOvalInRect
BezierPathFillOvalInRect

Bundle

new:
BundleMain = BundleRef
BundleWithURL(CFURLRef url) = BundleRef
BundleURLForResource(BundleRef bundle, CFStringRef name, CFStringRef extension, CFStringRef subdirectory) = CFURLRef
BundleURLForResource(BundleRef bundle, CFStringRef name) = CFURLRef
BundleResourceURL(BundleRef bundle) = CFURLRef
BundleExecutableRL(BundleRef bundle) = CFURLRef
BundleURL(BundleRef bundle) = CFURLRef
BundleIdentifier(BundleRef bundle) = CFStringRef
BundleInfoDictionary(BundleRef bundle) = CFDictionaryRef
BundleObjectForInfoDictionaryKey(BundleRef bundle, CFStringRef key) = CTypeRef

Button

new:
ButtonType(NSInteger tag, NSButtonType type)
ButtonGetPeriodicDelay(NSInteger tag, float *delay, float *interval)
ButtonAlternateTitle(NSInteger tag) = CFStringRef
ButtonAttributedTitle(NSInteger tag) = CFAttributedStringRef
ButtonAttributedAlternateTitle(NSInteger tag) = CFAttributedStringRef
ButtonSetTitle(NSInteger tag, CFStringRef title)
ButtonSound(NSInteger tag) = SoundRef
ButtonSetSoundNamed(NSInteger tag, CFStringRef name)
ButtonImage(NSInteger tag) = ImageRef
ButtonAlternateImage(NSInteger tag) = ImageRef
ButtonImagePosition(NSInteger tag) = NSCellImagePosition
ButtonIsBordered(NSInteger tag) = Boolean
ButtonIsTransparent(NSInteger tag) = Boolean
ButtonBezelStyle(NSInteger tag) = NSBezelStyle
ButtonSetBezelStyle(NSInteger tag, NSBezelStyle style)
ButtonBezelColor(NSInteger tag) = ColorRef// macOS 10.12.1+
ButtonSetBezelColor(NSInteger tag, ColorRef col)// macOS 10.12.1+
ButtonShowsBorderOnlyWhileMouseInside(NSInteger tag) = Boolean
ButtonImageHugsTitle(NSInteger tag) = Boolean// macOS 10.12+
ButtonSetImageHugsTitle(NSInteger tag, Boolean flag)// macOS 10.12+
ButtonImageScaling(NSInteger tag) = NSImageScaling
ButtonAllowsMixedState(NSInteger tag) = Boolean
ButtonSetState(NSInteger tag, NSCellStateValue state)
ButtonKeyEquivalent(NSInteger tag) = CFStringRef
ButtonKeyEquivalentModifierMask(NSInteger tag) = NSUInteger
ButtonImageDimsWhenDisabled(NSInteger tag) = Boolean

change:
ButtonSetSound(NSInteger tag, **SoundRef snd**)

CharacterSet

new:
CharacterSetAlphanumeric = CFCharacterSetRef
CharacterSetCapitalizedLetter = CFCharacterSetRef
CharacterSetControl = CFCharacterSetRef
CharacterSetDecimalDigit = CFCharacterSetRef
CharacterSetDecomposable = CFCharacterSetRef
CharacterSetIllegal = CFCharacterSetRef
CharacterSetLetter = CFCharacterSetRef
CharacterSetLowercaseLetter = CFCharacterSetRef
CharacterSetNewline = CFCharacterSetRef
CharacterSetNonBase = CFCharacterSetRef
CharacterSetPunctuation = CFCharacterSetRef

CharacterSetSymbol = CFCharacterSetRef
CharacterSetUppercaseLetter = CFCharacterSetRef
CharacterSetWhitespaceAndNewline = CFCharacterSetRef
CharacterSetWhitespace = CFCharacterSetRef
CharacterSetWithCharactersInString(CFStringRef string) = CFCharacterSetRef
CharacterSetWithRange(CFRange range) = CFCharacterSetRef
CharacterSetInvertedSet(CFCharacterSetRef set) = CFCharacterSetRef

Color

new:
ColorWithAlphaComponent(ColorRef col, CGFloat alpha) = ColorRef
ColorGetCMYK(ColorRef col, CGFloat *c, CGFloat *m, CGFloat *y, CGFloat *k, CGFloat *a)
ColorGetHSB(ColorRef col, CGFloat *h, CGFloat *s, CGFloat *b, CGFloat *a)
ColorGetRGB(ColorRef col, CGFloat *r, CGFloat *g, CGFloat *b, CGFloat *a)
ColorGetWhite(ColorRef col, CGFloat *w, CGFloat *a)
ColorNumberOfComponents(ColorRef col) = NSInteger
ColorGetComponents(ColorRef col, CGFloat *components)

Control

new:
ControlDrawWithExpansionFrameInView(NSInteger ctrlTag, CGRect contentFrame, NSInteger vwTag)// macOS 10.10+
ControlAllowsExpansionToolTips(NSInteger tag) = Boolean// macOS 10.8+
ControlSetAllowsExpansionToolTips(NSInteger tag, Boolean flag)// macOS 10.8+
ControlIgnoresMultiClick(NSInteger tag) = Boolean
ControlSetIgnoresMultiClick(NSInteger tag, Boolean flag)

Cursor

new:
CursorWithImage(ImageRef image, CGPoint hotSpot) = CursorRef
CursorImage(CursorRef cursor) = ImageRef
CursorHotSpot(CursorRef cursor) = CGPoint
CursorPop
CursorPush(CursorRef cursor)
CursorSet(CursorRef cursor)
CursorCurrent = CursorRef
CursorCurrentSystem = CursorRef
CursorArrow = CursorRef
CursorContextualMenu = CursorRef
CursorClosedHand = CursorRef
CursorCrosshair = CursorRef
CursorDisappearingItem = CursorRef
CursorDragCopy = CursorRef
CursorDragLink = CursorRef
CursorIBeam = CursorRef
CursorOpenHand = CursorRef
CursorOperationNotAllowed = CursorRef
CursorPointingHand = CursorRef
CursorResizeDown = CursorRef
CursorResizeLeft = CursorRef
CursorResizeRight = CursorRef
CursorResizeUp = CursorRef
CursorResizeUpDown = CursorRef
CursorIBeamForVerticalLayout = CursorRef

DockTile

new:
DockTileContentView(DockTileRef tile) = ViewRef
DockTileSize(DockTileRef tile) = CGSize
DockTileShowsApplicationBadge(DockTileRef tile) = Boolean
DockTileSetShowsApplicationBadge(DockTileRef tile, Boolean flag)
DockTileBadgeLabel(DockTileRef tile) = CFStringRef
DockTileSetBadgeLabel(DockTileRef tile, CFStringRef label)
DockTileDisplay(DockTileRef tile)

Font

new:
FontSet(FontRef font)

FontWithDescriptor(FontDescriptorRef descriptor, CGFloat size) = FontRef

FontDescriptor

new:
FontDescriptorWithAttributes(CFDictionaryRef attributes) = FontDescriptorRef

Image

new:
ImageByReferencingURL(CFURLRef url) = ImageRef
ImageLockFocus(ImageRef ref)
ImageLockFocusFlipped(ImageRef ref, Boolean flag)
ImageUnlockFocus(ImageRef ref)

Menu

new:
MenuItemArray(NSInteger menuIndex) = CFArrayRef

MenuItem

new:
MenuItemAttributedTitle(NSInteger menuIndex, NSInteger itemIndex) = CFAttributedStringRef
MenuItemSetAttributedTitle(NSInteger menuIndex, NSInteger itemIndex, CFAttributedStringRef title)

Object

new:
ObjectPerformSelectorOnMainThread(CTypeRef obj, CFStringRef selector, CTypeRef withObj, Boolean waitUntilDone)
ObjectPerformSelectorInBackground(CTypeRef obj, CFStringRef selector, CTypeRef withObj)
ObjectCallFunctionOnMainThread(NSInteger objTag, ptr fnAddress, CTypeRef withObj, Boolean waitUntilDone)
ObjectCallFunctionInBackground(ptr fnAddress, CTypeRef withObj)

OperationQueue

new:
OperationQueueInit = OperationQueueRef
OperationQueueAddOperation(OperationQueueRef queue, ptr fnAddress, CTypeRef obj)

OutlineView

new:
OutlineViewWithTag(NSInteger tag) = OutlineViewRef
OutlineViewIsItemExpandable(NSInteger tag, OVItemRef item) = Boolean
OutlineViewIsItemExpanded(NSInteger tag, OVItemRef item) = Boolean
OutlineViewExpandItem(NSInteger tag, OVItemRef item)
OutlineViewExpandItemAndChildren(NSInteger tag, OVItemRef item, Boolean expandChildren)
OutlineViewCollapseItem(NSInteger tag, OVItemRef item)
OutlineViewCollapseItemAndChildren(NSInteger tag, OVItemRef item, Boolean collapseChildren)
OutlineViewReloadItem(NSInteger tag, OVItemRef item)
OutlineViewReloadItemAndChildren(NSInteger tag, OVItemRef item, Boolean reloadChildren)
OutlineViewItemAtRow(NSInteger tag, NSInteger row) = OVItemRef
OutlineViewRowForItem(NSInteger tag, OVItemRef item) = NSInteger
OutlineViewLevelForItem(NSInteger tag, OVItemRef item) = NSInteger
OutlineViewLevelForRow(NSInteger tag, NSInteger row) = NSInteger
OutlineViewParentForItem(NSInteger tag, OVItemRef item) = OVItemRef
OutlineViewChildIndexForItem(NSInteger tag, OVItemRef item) = NSInteger
OutlineViewChildOfItem(NSInteger tag, NSInteger index, OVItemRef item) = OVItemRef
OutlineViewNumberOfChildrenOfItem(NSInteger tag, OVItemRef item) = NSInteger

Player

new:
PlayerWithURL(CFURLRef url) = PlayerRef

PlayerView

new:
PlayerViewWithTag(NSInteger tag) = PlayerViewRef
PlayerViewSetPlayer(NSInteger tag, PlayerRef player)

Popover

new:
PopoverBehavior(NSInteger tag) = NSPopoverBehavior
PopoverSetBehavior(NSInteger tag, NSPopoverBehavior behavior)

PopoverPositioningRect(NSInteger tag) = CGRect
PopoverSetPositioningRect(NSInteger tag, CGRect rect)
PopoverAnimates(NSInteger tag) = Boolean
PopoverSetAnimates(NSInteger tag, Boolean flag)
PopoverContentSize(NSInteger tag) = CGSize
PopoverSetContentSize(NSInteger tag, CGSize size)
PopoverIsDetached(NSInteger tag) = Boolean// macOS 10.10+

PopUpButton

new:
PopUpButtonMenu(NSInteger tag) = CocoaMenuRef
PopUpButtonSetMenuItemIndex(NSInteger tag, NSInteger menuItemIndex)
PopUpButtonArrowPosition(NSInteger tag) = NSPopUpArrowPosition
PopUpButtonSetArrowPosition(NSInteger tag, NSPopUpArrowPosition position)
PopUpButtonItemArray(NSInteger tag) = CFArrayRef

change:
PopUpButtonSetMenu(NSInteger tag, **CocoaMenuRef menu**)

RunningApplication

new:
RunningApplicationIcon(RunningApplicationRef ra) = ImageRef

Screen

new:
ScreenMainScreen = ScreenRef
ScreenDeepestScreen = ScreenRef
ScreenScreens = CFArrayRef
ScreenDepth(ScreenRef scrn) = NSWindowDepth
ScreenFrame(ScreenRef scrn) = CGRect
ScreenDeviceDescription(ScreenRef scrn) = CFDictionaryRef
ScreenVisibleFrame(ScreenRef scrn) = CGRect
ScreenCanRepresentDisplayGamut(ScreenRef scrn, NSInteger gamut) = Boolean// macOS 10.12+
ScreenScreensHaveSeparateSpaces = Boolean// macOS 10.9+
ScreenBackingAlignedRect(ScreenRef scrn, CGRect rect, NSTextAlignmentOptions options) = CGRect
ScreenBackingScaleFactor(ScreenRef scrn) = CGFloat
ScreenConvertRectFromBacking(ScreenRef scrn, CGRect rect) = CGRect
ScreenConvertRectToBacking(ScreenRef scrn, CGRect rect) = CGRect
ScreenMaximumExtendedDynamicRangeColorComponentValue(ScreenRef scrn) = CGFloat// macOS 10.11+

ScrollView

new:
ScrollViewAddFloatingSubview(NSInteger scrollViewTag, NSInteger subviewTag, NSEventGestureAxis axis)// macOS 10.9+
ScrollViewReflectScrolledClipView(NSInteger tag)
ScrollViewScrollToPoint(NSInteger tag, CGPoint pt)

typo fix:
ScrollViewSetHorizontalLineScroll was ScrollViewSetLHorizontalLineScroll

SegmentedControl

new:
SegmentedControlImageScaling(NSInteger tag, NSInteger index) = NSImageScaling
SegmentedControlSetImageScaling(NSInteger tag, NSInteger index, NSImageScaling scaling)

String

new:
StringComponentsSeparatedByString(CFStringRef string, CFStringRef separator) = CFArrayRef
StringByRemovingPercentEncoding(CFStringRef string) = CFStringRef// macOS 10.9+

TableView

new:
TableViewBeginUpdates(NSInteger tag)
TableViewEndUpdates(NSInteger tag)
TableViewHideRow(NSInteger tag, NSInteger rowIndex, NSTableViewAnimationOptions animation)// macOS 10.11+
TableViewHideRows(NSInteger tag, IndexSetRef indexes, NSTableViewAnimationOptions animation)// macOS 10.11+
TableViewUnhideRow(NSInteger tag, NSInteger rowIndex, NSTableViewAnimationOptions animation)// macOS 10.11+
TableViewUnhideRows(NSInteger tag, IndexSetRef indexes, NSTableViewAnimationOptions animation)// macOS 10.11+

TableViewHiddenRowIndexes(NSInteger tag) = IndexSetRef// macOS 10.11+

Text

new:
TextShowGuessPanel(NSInteger tag)

Toolbar

new:
ToolbarsVisible(NSInteger tag) = Boolean
ToolbarRunCustomizationPalette(NSInteger tag)
ToolbarCustomizationPalettelsRunning(NSInteger tag) = Boolean

View

new:
ViewsFlipped(NSInteger tag) = Boolean
ViewAutoresizingMask(NSInteger tag) = NSAutoresizingMaskOptions

Window

new:
WindowAlphaValue(NSInteger tag) = CGFloat
WindowBackgroundColor(NSInteger tag) = ColorRef
WindowCanHide(NSInteger tag) = Boolean
WindowHidesOnDeactivate(NSInteger tag) = Boolean
WindowHasShadow(NSInteger tag) = Boolean
WindowPreventsApplicationTerminationWhenModal(NSInteger tag) = Boolean
WindowNumber(NSInteger tag) = NSInteger
WindowNumbersWithOptions(NSWindowNumberListOptions options) = CFArrayRef// array of CFNumbers
WindowContentRectForFrameRectAndStyle(CGRect r, NSInteger styleMask) = CGRect
WindowFrameRectForContentRectAndStyle(CGRect r, NSInteger styleMask) = CGRect
WindowMinFrameWidthWithTitle(CFStringRef title, NSInteger styleMask) = CGFloat
WindowAspectRatio(NSInteger tag) = CGSize
WindowShowsResizeIndicator(NSInteger tag) = Boolean
WindowResizeIncrements(NSInteger tag) = CGSize
WindowContentMinSize(NSInteger tag) = CGSize
WindowContentSize(NSInteger tag) = CGSize
WindowContentSize(NSInteger tag) = CGSize
WindowContentMaxSize(NSInteger tag) = CGSize
WindowContentResizeIncrements(NSInteger tag) = CGSize
WindowMaxFullScreenContentSize(NSInteger tag) = CGSize// macOS 10.11+
WindowMinFullScreenContentSize(NSInteger tag) = CGSize// macOS 10.11+
WindowLevel(NSInteger tag) = NSWindowLevel
WindowFrameAutosaveName(NSInteger tag) = CFStringRef
WindowIsExcludedFromWindowsMenu(NSInteger tag) = Boolean
WindowTitlebarAppearsTransparent(NSInteger tag) = Boolean// macOS 10.10
WindowAutorecalculatesKeyViewLoop(NSInteger tag) = Boolean
WindowAcceptsMouseMovedEvents(NSInteger tag) = Boolean
WindowSetAcceptsMouseMovedEvents(NSInteger tag, Boolean flag)
WindowIgnoresMouseEvents(NSInteger tag) = Boolean
WindowSetIgnoresMouseEvents(NSInteger tag, Boolean flag)
WindowTitleVisibility(NSInteger tag) = NSWindowTitleVisibility// macOS 10.10+
WindowIsMovableByWindowBackground(NSInteger tag) = Boolean
WindowIsMovable(NSInteger tag) = Boolean
WindowMiniwindowImage(NSInteger tag) = ImageRef
WindowMiniwindowTitle(NSInteger tag) = CFStringRef
WindowDockTile(NSInteger tag) = DockTileRef

New demos

- Application
 - AppDockTile
- EventHandlers
 - App
 - Menu
 - Window
- Graphics
 - Gradient
- Graphics/BezierPath

- DotView
- DrawImage
- DrawInScrollingView
- RotatedRect
- Miscellaneous/Threads
 - Simple Background Thread 1
 - Simple Background Thread 2
 - Simple Background Thread 3
- Miscellaneous/Threads/Thread Alternatives
 - OperationQueue
- Views and Controls/ImageView
 - DrawIntolImage
- Views and Controls/OutlineView
 - Single column
 - Two column
- Views and Controls
 - PlayerView
- Views and Controls/PopUpButton
 - ModifyPopUpButton 1
 - ModifyPopUpButton 2
- Views and Controls/TableView/Images
 - Images 1
 - Images 2
 - Images 3
- Windows and Panels/Window
 - DockTile
- Windows and Panels/Alert
 - Alert suppression button

FB 5.7.113 requires OS X 10.7+ for CocoaUI apps. Non-CocoaUI require 10.6

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.112]

[01] Replacement call, PMPrintSettingsSetJobName(), added to Tlhx Printing.incl.
PMSetJobNameCFString, now removed by Apple, commented out.
[02] Long, tight loop in FB file I/O (PRINT#, READ, WRITE etc.) exhausted file descriptors,
so I/O fails. Note: see list post on 28-Feb-2018 with subject "File I/O update and news
regarding Carbon/Cocoa" for more information on nature of the issue and the implications of the
fix.
[03] Built application unexpected quits. An explanation follows:

Testing new file I/O highlighted another issue. The app would unexpectedly quit. Bernie did
some good investigative work and he remembered a quitting issue right after a file open/save
panel's display. *Bottom line: mixing carbon events and cocoa events was causing the quit
problem.* CocoaUI code was not involved. Two things discovered:

a. Removing a call to NSApplicationLoad fixes it BUT that isn't a solution for those
needing Cocoa in Carbon apps. See #2b for that.

b. Enclosing HandleEvents in a 'do/until' loop also fixes it:

The following snippet(courtesy Bernie) demonstrates the bug and fix:

```
//-----  
toolbox fn NSApplicationLoad = Boolean  
fn NSApplicationLoad // removing this call fixes it, but that's not good for Cocoa in Carbon  
apps.  
  
local fn ShowAlert  
  '~'1  
  BeginCCode  
    NSAlert *alert = [[NSAlert alloc] init];  
    [alert setAlertStyle:NSWarningAlertStyle];  
    [alert setMessageText:@"Hello"];  
    [alert setInformativeText:@"Something happened."];
```

```

    [alert runModal];
EndC
end fn

void local fn DoMenu
    '~'1
    if ( menu(_menuID) == 1 and menu(_itemID) == 1 ) then fn ShowAlert
end fn

menu 1,0,1,@"File"
menu 1,1,1,@"Show Alert/A"

on menu fn DoMenu

// enclosing HandleEvents in the do until loop also fixes it
//do
HandleEvents
//until gFBQuit
//-----

```

CocoaUI changes

Likely causes of failure this time:

1. All widget functions that took RGB values (TextFieldSetTextRGB, WindowSetBackgroundRGB, etc.) have been removed. Make a ColorRef with ColorWithRGB and use XxxSetColor instead.
2. Widget functions that took a CGImageRef have been removed. Convert the CGImage to an ImageRef with ImageWithCGImage and use XxxSetImage instead.
3. String function extra param(s).
StringWriteToURL(CFStringRef string, CFURLRef url, Boolean atomically, NSStringEncoding enc, ErrorRef *err) = Boolean // err can be NULL
StringWithContentsOfURL(CFURLRef url, NSStringEncoding enc, ErrorRef *err) = CFStringRef // err can be NULL

Alert

new:
AlertSetIconNamed(NSInteger tag, CFStringRef imageName)

Application

new:
AppOrderFrontColorPanel
AppOrderFrontCharacterPalette

fix:
_applicationShouldTerminate now works for application events

Array

new:
ArrayContainsObject(CFArrayRef array, CTypeRef obj) = Boolean
ArrayCount(CFArrayRef array) = NSUInteger
ArrayObjectAtIndex(CFArrayRef array, NSUInteger index) = CTypeRef
ArrayObjectsAtIndexes(CFArrayRef array, IndexSetRef indexes) = CFArrayRef
ArrayIndexForObject(CFArrayRef array, CTypeRef obj) = NSUInteger
ArrayIndexForObjectInRange(CFArrayRef array, CTypeRef obj, CFRange range) = NSUInteger
ArrayComponentsJoinedByString(CFArrayRef array, CFStringRef separator) = CFStringRef
ArrayAddObject(CFMutableArrayRef array, CTypeRef obj)
ArrayAddObjectsFromArray(CFMutableArrayRef array, CFArrayRef otherArray)
ArrayInsertObjectAtIndex(CFMutableArrayRef array, CTypeRef obj, NSUInteger index)
ArrayInsertObjectsAtIndexes(CFMutableArrayRef array, CFArrayRef objects, IndexSetRef indexes)
ArrayRemoveAllObjects(CFMutableArrayRef array)
ArrayRemoveLastObject(CFMutableArrayRef array)
ArrayRemoveObject(CFMutableArrayRef array, CTypeRef obj)
ArrayRemoveObjectInRange(CFMutableArrayRef array, CTypeRef obj, CFRange range)
ArrayRemoveObjectAtIndex(CFMutableArrayRef array, NSUInteger index)

ArrayRemoveObjectsAtIndexes(CFMutableArrayRef array, IndexSetRef indexes)
ArrayRemoveObjectsInArray(CFMutableArrayRef array, CFArrayRef objects)
ArrayRemoveObjectsInRange(CFMutableArrayRef array, CFRange range)
ArrayReplaceObjectAtIndex(CFMutableArrayRef array, CTypeRef obj, NSUInteger index)
ArrayReplaceObjectsAtIndexes(CFMutableArrayRef array, CFArrayRef objects, IndexSetRef indexes)
ArraySetArray(CFMutableArrayRef array, CFArrayRef otherArray)
ArrayExchangeObjects(CFMutableArrayRef array, NSUInteger index1, NSUInteger index2)

BezierPath

new:
BezierPathGetLineDash(BezierPathRef path, CGFloat *pattern, NSInteger *count, CGFloat *phase)
BezierPathDefaultWindingRule = NSWindingRule
BezierPathDefaultLineCapStyle = NSLineCapStyle
BezierPathDefaultLineJoinStyle = NSLineJoinStyle
BezierPathDefaultLineWidth = CGFloat
BezierPathDefaultMiterLimit = CGFloat
BezierPathDefaultFlatness = CGFloat
BezierPathControlPointBounds(BezierPathRef path) = CGRect

Box

changes:
BoxSetBorderRGB & BoxSetFillRGB removed. Use BoxSetBorderColor & BoxSetFillColor instead.

Button

new:
ButtonSetImageNamed(NSInteger tag, CFStringRef imageName)
ButtonSetAlternateImageNamed(NSInteger tag, CFStringRef imageName)

changes:
ButtonSetTitleRGB & ButtonSetBackgroundRGB removed. Use ButtonSetColor & ButtonSetBackgroundColor instead.
ButtonIsEnabled removed. Use ControlsEnabled instead.

Color

new:
// system colors
ColorAlternateSelectedControl = ColorRef
ColorAlternateSelectedControlText = ColorRef
ColorForControlTint(NSControlTint tint) = ColorRef
ColorControlBackground = ColorRef
ColorControl = ColorRef
ColorControlAlternatingRowBackground = CFArrayRef
ColorControlHighlight = ColorRef
ColorControlShadow = ColorRef
ColorControlDarkShadow = ColorRef
ColorControlText = ColorRef
ColorCurrentControlTint = NSControlTint
ColorDisabledControlText = ColorRef
ColorGrid = ColorRef
ColorHeader = ColorRef
ColorHeaderText = ColorRef
ColorHighlight = ColorRef
ColorKeyboardFocusIndicator = ColorRef
ColorKnob = ColorRef
ColorScrollBar = ColorRef
ColorScrubberTexturedBackground = ColorRef// macOS 10.11+
ColorSecondarySelectedControl = ColorRef
ColorSelectedControl = ColorRef
ColorSelectedControlText = ColorRef
ColorSelectedMenuItem = ColorRef
ColorSelectedMenuItemText = ColorRef
ColorSelectedTextBackground = ColorRef
ColorSelectedText = ColorRef
ColorSelectedKnob = ColorRef
ColorShadow = ColorRef
ColorSystemBlue = ColorRef// macOS 10.11+
ColorSystemBrown = ColorRef// macOS 10.11+
ColorSystemGray = ColorRef// macOS 10.11+

ColorSystemGreen = ColorRef// macOS 10.11+
ColorSystemOrange = ColorRef// macOS 10.11+
ColorSystemPink = ColorRef// macOS 10.11+
ColorSystemPurple = ColorRef// macOS 10.11+
ColorSystemRed = ColorRef// macOS 10.11+
ColorSystemYellow = ColorRef// macOS 10.11+
ColorTextBackground = ColorRef
ColorText = ColorRef
ColorWindowBackground = ColorRef
ColorWindowFrame = ColorRef
ColorWindowFrameText = ColorRef
ColorUnderPageBackground = ColorRef// macOS 10.8+

// system label colors
ColorLabel = ColorRef// macOS 10.10+
ColorSecondaryLabel = ColorRef// macOS 10.10+
ColorTertiaryLabel = ColorRef// macOS 10.10+
ColorQuaternaryLabel = ColorRef// macOS 10.10+

// pattern
ColorWithPatternImage(ptr image) = ColorRef
ColorPatternImage(ColorRef col) = ImageRef

// retrieving individual components
ColorAlphaComponent(ColorRef col) = CGFloat
ColorWhiteComponent(ColorRef col) = CGFloat
ColorRedComponent(ColorRef col) = CGFloat
ColorGreenComponent(ColorRef col) = CGFloat
ColorBlueComponent(ColorRef col) = CGFloat
ColorCyanComponent(ColorRef col) = CGFloat
ColorMagentaComponent(ColorRef col) = CGFloat
ColorYellowComponent(ColorRef col) = CGFloat
ColorBlackComponent(ColorRef col) = CGFloat
ColorHueComponent(ColorRef col) = CGFloat
ColorSaturationComponent(ColorRef col) = CGFloat
ColorBrightnessComponent(ColorRef col) = CGFloat

changes:
ColorSetWithCGColor, ColorSetFillWithCGColor, ColorSetStrokeWithCGColor removed. Use ColorSet, ColorSetFill, ColorSetStroke instead.

ColorPanel

change:
ColorPanelColor = **ColorRef**
ColorPanelSetRGB removed. Use ColorPanelSetColor instead

ColorWell

new:
ColorWellIsBordered(NSInteger tag) = Boolean
ColorWellSetBordered(NSInteger tag, Boolean flag)
ColorWellIsActive(NSInteger tag) = Boolean

changes:
ColorWellColor(NSInteger tag) = **ColorRef**
ColorWellSetRGB removed. Use ColorWellSetColor instead

Control

new:
ControlSetTargetAction(NSInteger ctrlTag, NSInteger targetTag, CFStringRef actionName)

Cursor

new:
CursorSetImageNamed(CFStringRef imageName, CGPoint hotSpot)

DatePicker

changes:

DatePickerSetTextRGB & DatePickerSetBackgroundRGB removed. Use DatePickerSetTextColor & DatePickerSetBackgroundColor instead.

Dictionary

new:
DictionaryWithObjectsForKeys(CFArrayRef objects, CFArrayRef keys) = CFDictionaryRef
DictionaryCount(CFDictionaryRef dict) = NSUInteger
DictionaryIsEqual(CFDictionaryRef dict, CFDictionaryRef otherDict) = Boolean
DictionaryAllKeys(CFDictionaryRef dict) = CFArrayRef
DictionaryAllKeysForObject(CFDictionaryRef dict, CTypeRef obj) = CFArrayRef
DictionaryAllValues(CFDictionaryRef dict) = CFArrayRef
DictionaryObjectForKey(CFDictionaryRef dict, CFStringRef key) = CTypeRef
DictionarySetObject(CFMutableDictionaryRef dict, CTypeRef obj, CFStringRef key)
DictionaryAddEntriesFromDictionary(CFMutableDictionaryRef dict, CFDictionaryRef otherDict)
DictionarySetDictionary(CFMutableDictionaryRef dict, CFDictionaryRef otherDict)
DictionaryRemoveObjectForKey(CFMutableDictionaryRef dict, CFStringRef key)
DictionaryRemoveAllObjects(CFMutableDictionaryRef dict)
DictionaryRemoveObjectsForKeys(CFMutableDictionaryRef dict, CFArrayRef keys)

Error

new:
ErrorCode(ErrorRef err) = NSInteger
ErrorDomain(ErrorRef err) = CFStringRef
ErrorUserInfo(ErrorRef err) = CFDictionaryRef

Gradient

new:
GradientWithStartEndColors(ColorRef startColor, ColorRef endColor) = GradientRef
GradientWithColors(CFArrayRef colors) = GradientRef
GradientDrawFromPoint(GradientRef ref, CGPoint pt1, CGPoint pt2, NSGradientDrawingOptions options)
GradientDrawFromCenter(GradientRef ref, CGPoint startCenter, CGFloat startRadius, CGPoint endCenter, CGFloat endRadius, NSGradientDrawingOptions options)
GradientDrawInRect(GradientRef ref, CGRect rect, CGFloat angle)
GradientDrawInBezierPath(GradientRef ref, BezierPathRef path, CGFloat angle)
GradientDrawInRectRelativeCenterPosition(GradientRef ref, CGRect rect, CGPoint position)
GradientDrawInBezierPathRelativeCenterPosition(GradientRef ref, BezierPathRef path, CGPoint position)
GradientNumberOfColorStops(GradientRef ref) = NSInteger
GradientDrawFromPointWithStartEndColors(CGPoint pt1, CGPoint pt2, ColorRef startColor, ColorRef endColor, NSGradientDrawingOptions options)
GradientDrawFromCenterWithStartEndColors(CGPoint startCenter, CGFloat startRadius, CGPoint endCenter, CGFloat endRadius, ColorRef startColor, ColorRef endColor, NSGradientDrawingOptions options)
GradientDrawInRectWithStartEndColors(CGRect rect, CGFloat angle, ColorRef startColor, ColorRef endColor)
GradientDrawInBezierPathWithStartEndColors(BezierPathRef path, CGFloat angle, ColorRef startColor, ColorRef endColor)
GradientDrawInRectRelativeCenterPositionWithStartEndColors(CGRect rect, CGPoint position, ColorRef startColor, ColorRef endColor)
GradientDrawInBezierPathRelativeCenterPositionWithStartEndColors(BezierPathRef path, CGPoint position, ColorRef startColor, ColorRef endColor)

Graphics

new:
GraphicsShowAnimationEffect(NSAnimationEffect animationEffect, CGPoint centerLocation, CGSize size, ptr didEndCallback, ptr contextInfo)

GraphicsContext

new:
GraphicsContextSaveGraphicsState
GraphicsContextRestoreGraphicsState

Image

new:
ImageWithCGImage(CGImageRef image, CGSize size) = ImageRef

ImageView

new:
ImageViewSetImageNamed(NSInteger tag, CFStringRef imageName)
ImageViewImage(NSInteger tag) = ImageRef

changes:

The **imageview** statement *image* param value can now be any of the following:

- The name of a resource image
- The path to a resource image
- An ImageRef (NSImage)

ImageViewCGImage removed. Use ImageViewImage instead.

ImageViewSetCGImage removed. Use ImageViewSetImage instead.

LevelIndicator

new:

LevelIndicatorSetImageNamed(NSInteger tag, CFStringRef imageName)

LevelIndicatorSetRatingPlaceholderImageNamed(NSInteger tag, CFStringRef imageName)

PathUtilities

new:

NSUserName = CFStringRef

NSFullUserName = CFStringRef

NSHomeDirectory = CFStringRef

NSHomeDirectoryForUser(CFStringRef userName) = CFStringRef

NSTemporaryDirectory = CFStringRef

NSOpenStepRootDirectory = CFStringRef

Printer

new:

PrinterWithName(CFStringRef name) = PrinterRef

PrinterWithType(NSPrinterTypeName type) = PrinterRef

PrinterNames = CFArrayRef

PrinterTypes = CFArrayRef

PrinterName(PrinterRef printer) = CFStringRef

PrinterType(PrinterRef printer) = NSPrinterTypeName

PrinterPageSizeForPaper(PrinterRef printer, NSPrinterPaperName paperName) = CGSize

PrinterLanguageLevel(PrinterRef printer) = NSInteger

PrintInfo

new:

PrintInfoShared = PrintInfoRef

PrintInfoLeftMargin = CGFloat

PrintInfoSetLeftMargin(CGFloat value)

PrintInfoTopMargin = CGFloat

PrintInfoSetTopMargin(CGFloat value)

PrintInfoRightMargin = CGFloat

PrintInfoSetRightMargin(CGFloat value)

PrintInfoBottomMargin = CGFloat

PrintInfoSetBottomMargin(CGFloat value)

PrintInfoImageablePageBounds = CGRect

PrintInfoOrientation = NSPaperOrientation

PrintInfoSetOrientation(NSPaperOrientation orientation)

PrintInfoPaperName = NSPrinterPaperName

PrintInfoSetPaperName(NSPrinterPaperName string)

PrintInfoLocalizedPaperName = CFStringRef

PrintInfoPaperSize = CGSize

PrintInfoSetPaperSize(CGSize size)

PrintInfoHorizontalPagination = NSPrintingPaginationMode

PrintInfoSetHorizontalPagination(NSPrintingPaginationMode value)

PrintInfoVerticalPagination = NSPrintingPaginationMode

PrintInfoSetVerticalPagination(NSPrintingPaginationMode value)

PrintInfoIsHorizontallyCentered = Boolean

PrintInfoSetHorizontallyCentered(Boolean flag)

PrintInfoIsVerticallyCentered = Boolean

PrintInfoSetVerticallyCentered(Boolean flag)

PrintInfoPrinter = PrinterRef

PrintInfoSetPrinter(PrinterRef printer)

PrintInfoJobDisposition = NSPrintJobDispositionValue

PrintInfoSetJobDisposition(NSPrintJobDispositionValue value)

PrintInfoSetUpPrintOperationDefaultValues

PrintInfoDictionary = CFDictionaryRef

PrintInfoIsSelectionOnly = Boolean

PrintInfoSetSelectionOnly(Boolean flag)
PrintInfoScalingFactor = CGFloat
PrintInfoSetScalingFactor(CGFloat value)
PrintInfoPrintSettings = CFMutableDictionaryRef

PrintOperation

new:
PrintOperationEPSWithViewToData(NSInteger viewTag, CGRect rect, CFMutableDataRef data) = PrintOperationRef
PrintOperationEPSWithViewToDataPrintInfo(NSInteger viewTag, CGRect rect, CFMutableDataRef data, PrintInfoRef printInfo) = PrintOperationRef
PrintOperationEPSWithViewToPath(NSInteger viewTag, CGRect rect, CFStringRef path, PrintInfoRef printInfo) = PrintOperationRef
PrintOperationPDFWithViewToData(NSInteger viewTag, CGRect rect, CFMutableDataRef data) = PrintOperationRef
PrintOperationPDFWithViewToDataPrintInfo(NSInteger viewTag, CGRect rect, CFMutableDataRef data, PrintInfoRef printInfo) = PrintOperationRef
PrintOperationPDFWithViewToPath(NSInteger viewTag, CGRect rect, CFStringRef path, PrintInfoRef printInfo) = PrintOperationRef
PrintOperationWithView(NSInteger viewTag) = PrintOperationRef
PrintOperationWithViewPrintInfo(NSInteger viewTag, PrintInfoRef printInfo) = PrintOperationRef
PrintOperationCurrent = PrintOperationRef
PrintOperationIsCopying(PrintOperationRef printOperation) = Boolean
PrintOperationPrintInfo(PrintOperationRef printOperation) = PrintInfoRef
PrintOperationSetPrintInfo(PrintOperationRef printOperation, PrintInfoRef printInfo)
PrintOperationView(PrintOperationRef printOperation) = NSInteger
PrintOperationPreferredRenderingQuality(PrintOperationRef printOperation) = NSPrintRenderingQuality
PrintOperationRun(PrintOperationRef printOperation) = Boolean
PrintOperationDeliverResult(PrintOperationRef printOperation) = Boolean
PrintOperationShowsPrintPanel(PrintOperationRef printOperation) = Boolean
PrintOperationSetShowsPrintPanel(PrintOperationRef printOperation, Boolean flag)
PrintOperationShowsProgressPanel(PrintOperationRef printOperation) = Boolean
PrintOperationSetShowsProgressPanel(PrintOperationRef printOperation, Boolean flag)
PrintOperationJobTitle(PrintOperationRef printOperation) = CFStringRef
PrintOperationSetJobTitle(PrintOperationRef printOperation, CFStringRef title)
PrintOperationPrintPanel(PrintOperationRef printOperation) = PrintPanelRef
PrintOperationSetPrintPanel(PrintOperationRef printOperation, PrintPanelRef printPanel)
PrintOperationContext(PrintOperationRef printOperation) = GraphicsContextRef
PrintOperationCreateContext(PrintOperationRef printOperation) = GraphicsContextRef
PrintOperationCurrentPage(PrintOperationRef printOperation) = NSInteger
PrintOperationPageRange(PrintOperationRef printOperation) = CFRange
PrintOperationPageOrder(PrintOperationRef printOperation) = NSPrintingPageOrder
PrintOperationSetPageOrder(PrintOperationRef printOperation, NSPrintingPageOrder pageOrder)
PrintOperationCanSpawnSeparateThread(PrintOperationRef printOperation) = Boolean
PrintOperationSetCanSpawnSeparateThread(PrintOperationRef printOperation, Boolean flag)

RunningApplication

new:
WorkspaceMenuBarOwningApplication = RunningApplicationRef

ScrollView

changes:
ScrollViewBackgroundColor(NSInteger tag) = **ColorRef**
ScrollViewSetBackgroundRGB removed. Use ScrollViewSetBackgroundColor instead.

SegmentedControl

new:
SegmentedControlSetSegmentWidth(NSInteger tag, NSInteger index, CGFloat value)
SegmentedControlSetSegmentImage(NSInteger tag, NSInteger index, ImageRef image)
SegmentedControlSetSegmentLabel(NSInteger tag, NSInteger index, CFStringRef string)
SegmentedControlSetSegmentMenu(NSInteger tag, NSInteger index, MenuRef menu)
SegmentedControlSetSegmentSelected(NSInteger tag, NSInteger index, Boolean flag)
SegmentedControlIsSegmentSelected(NSInteger tag, NSInteger index) = Boolean
SegmentedControlSetSegmentEnabled(NSInteger tag, NSInteger index, Boolean flag)
SegmentedControlIsSegmentEnabled(NSInteger tag, NSInteger index) = Boolean
SegmentedControlIndexOfSelectedItem(NSInteger tag) = NSInteger// macOS 10.13+
SegmentedControlSetSegmentDistribution(NSInteger tag, NSSegmentDistribution value)// macOS 10.13+
SegmentedControlSetSelectedSegmentBezelColor(NSInteger tag, ColorRef col)// macOS 10.12.1+
SegmentedControlSetAlignment(NSInteger tag, NSInteger index, NSTextAlignment value)// macOS 10.13+

SegmentedControlSetShowsMenuIndicator(NSInteger tag, NSInteger index, Boolean flag)// macOS 10.13+
SegmentedControlSetToolTip(NSInteger tag, NSInteger index, CFStringRef string)// macOS 10.13+

change:
SegmentedControlSetMenu changed to SegmentedControlSetSegmentMenuIndex

Shadow
new:
ShadowInit = ShadowRef
ShadowOffset(ShadowRef shadow) = CGSize
ShadowSetOffset(ShadowRef shadow, CGSize offset)
ShadowBlurRadius(ShadowRef shadow) = CGFloat
ShadowSetBlurRadius(ShadowRef shadow, CGFloat radius)
ShadowColor(ShadowRef shadow) = ColorRef
ShadowSetColor(ShadowRef shadow, ColorRef col)
ShadowSet(ShadowRef shadow)
ShadowWithAttributes(CGSize offset, CGFloat blurRadius, ColorRef col) = ShadowRef
ShadowSetWithAttributes(CGSize offset, CGFloat blurRadius, ColorRef col)

Stepper
change:
StepperSetContinuous removed. Use ControlSetContinuous instead.

String
new:
StringLength(CFStringRef string) = NSUInteger
StringCharacterAtIndex(CFStringRef string, NSUInteger index) = unichar
StringCaseInsensitiveCompare(CFStringRef string, CFStringRef otherString) = NSComparisonResult
StringCompare(CFStringRef string, CFStringRef otherString) = NSComparisonResult
StringHasPrefix(CFStringRef string, CFStringRef prefix) = Boolean
StringHasSuffix(CFStringRef string, CFStringRef suffix) = Boolean
StringIsEqualToString(CFStringRef string, CFStringRef otherString) = Boolean
StringDrawAtPoint(CFStringRef string, CGPoint pt, CFDictionaryRef attributes)
StringDrawInRect(CFStringRef string, CGRect rect, CFDictionaryRef attributes)
StringSizeWithAttributes(CFStringRef string, CFDictionaryRef attributes) = CGSize
StringLowercaseString(CFStringRef string) = CFStringRef
StringLocalizedLowercaseString(CFStringRef string) = CFStringRef// macOS 10.11+
StringUppercaseString(CFStringRef string) = CFStringRef
StringLocalizedUppercaseString(CFStringRef string) = CFStringRef// macOS 10.11+
StringCapitalizedString(CFStringRef string) = CFStringRef
StringLocalizedCapitalizedString(CFStringRef string) = CFStringRef// macOS 10.11+
StringDoubleValue(CFStringRef string) = double
StringFloatValue(CFStringRef string) = float
StringIntegerValue(CFStringRef string) = NSInteger
StringLongLongValue(CFStringRef string) = SInt64
StringBoolValue(CFStringRef string) = Boolean
StringPathWithComponents(CFArrayRef array) = CFStringRef
StringPathComponents(CFStringRef string) = CFArrayRef
StringLastPathComponent(CFStringRef string) = CFStringRef
StringPathExtension(CFStringRef string) = CFStringRef
StringByAbbreviatingWithTildeInPath(CFStringRef string) = CFStringRef
StringByAppendingPathComponent(CFStringRef string, CFStringRef component) = CFStringRef
StringByAppendingPathExtension(CFStringRef string, CFStringRef extension) = CFStringRef
StringByDeletingLastPathComponent(CFStringRef string) = CFStringRef
StringByDeletingPathExtension(CFStringRef string) = CFStringRef
StringByExpandingTildeInPath(CFStringRef string) = CFStringRef
StringAppendFormat(CFMutableStringRef string, CFStringRef format, ...)
StringAppendString(CFMutableStringRef string, CFStringRef otherString)
StringApplyTransform(CFMutableStringRef string, NSStringTransform transform, Boolean reverse, CFRange range, CFRange *resultingRange)// macOS 10.11+
StringDeleteCharacters(CFMutableStringRef string, CFRange range)
StringInsertString(CFMutableStringRef string, CFStringRef otherString, NSUInteger index)
StringReplaceCharacters(CFMutableStringRef string, CFStringRef otherString, CFRange range)
StringReplaceOccurrencesOfString(CFMutableStringRef string, CFStringRef targetString, CFStringRef replacementString, NSStringCompareOptions options, CFRange range)
StringSetString(CFMutableStringRef string, CFStringRef otherString)
StringWithContentsOfURLUsedEncoding(CFURLRef url, NSStringEncoding *enc, ErrorRef *err) = CFStringRef

changes:
StringWriteToURL now has an extra param, ErrorRef *err. Those not interested in details of errors may pass NULL for this param.
StringWithContentsOfURL now has two extra params:
 NSStringEncoding enc. See String.incl header for possible values.
 ErrorRef *err. Those not interested in details of errors may pass NULL for this param.

TableView

changes:
For convenience, the table data array for a single-column tableview may now optionally contain an array of strings rather than an array of dictionaries.

Text

change:
TextColor(NSInteger tag) = **ColorRef**

TextField

change:
TextFieldSetTextRGB & TextFieldSetBackgroundRGB removed. Use TextFieldSetColor & TextFieldSetBackgroundColor instead.

TextView

change:
TextViewSetBackgroundRGB removed. Use TextViewSetBackgroundColor instead.

ToolbarItem

change:
The *image* param value of the **toolobaritem** statement can now be the name of a resource image, the path to a resource image or an ImageRef (NSImage).

View

new:
ViewSubviews(NSInteger tag) = CFArrayRef
ViewTag(ViewRef view) = NSInteger

Window

new:
WindowOcclusionState(NSInteger tag) = NSWindowOcclusionState// macOS 10.9+
WindowSetMiniwindowImageNamed(NSInteger tag, CFStringRef imageName)
WindowSetMovableByWindowBackground(NSInteger tag, Boolean flag)
WindowSetMovable(NSInteger tag, Boolean flag)

change:
WindowSetBackgroundRGB removed. Use WindowSetBackgroundColor instead

Workspace

new:
WorkspacelconForFileAtURL(CFURLRef url) = ImageRef
WorkspacelconForFileType(CFStringRef type) = ImageRef
WorkspaceShowSearchResultsForQueryString(CFStringRef string)

20180213 FB 5.7.112

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.111]

FB 5.7.112 requires OS X 10.7+ for CocoaUI apps. Non-CocoaUI require 10.6

[01] FBtoC now accepts up to forty parameters inside a function call. The previous maximum was 16.

[02] _kHIWindowBitTextured added to standard headers inside Tlbx MacWindows.incl

CocoaUI changes

Please note the following:

1. Many of the more recent Util_xxxx.incl files are now integrated into CocoaUI. Those includes are no longer required because Tlxb CocoaUI.incl makes them available. Even though they are integrated, the function names have changed slightly. For example, instead of calling FM_TrashItemAtURL(url) and using the Util_FileManager.incl, call fn FileManagerTrashItemAtURL(url) and use only Tlxb CocoaUI.incl. Impacted Util_xxxx.incl files are shown below and are still available for those not using CocoaUI.

- Util_Array.incl
- Util_Dictionary.incl
- Util_FileManager.incl
- Util_Number.incl
- Util_PathUtilities.incl
- Util_RunningApplications.incl
- Util_String.incl
- Util_URL.incl
- Util_UserDefaults.incl
- Util_Workspace.incl

2. Because of the changes in #1, some apps will break and will need changes. The CocoaUI user will need to:

- Remove any old include statements (i.e. Util_FileManager.incl)
- Rename the function calls (FM_Xxx -> FileManagerXxxx, etc.).

3. A name change for function GraphicsCurrentCGContext. It is now GraphicsContextCurrentCGContext

4. Examples for the former Util_xxx.incl functions noted in #1 have been updated and copied to FB Examples/Cocoa/CocoaUI Demos

General

changes:

SInt32 and UInt32 types are now replaced by NSInteger and NSUInteger respectively.

Array

new:

```
ArrayWithCapacity( NSUInteger numItems ) = CFMutableArrayRef
ArrayWithContentsOfURL( CFURLRef url ) = CFArrayRef
ArrayWithObject( CTypeRef obj ) = CFArrayRef
ArrayWithObjects( CTypeRef obj, ... ) = CFArrayRef
ArrayFirstObject( CFArrayRef array ) = CTypeRef
ArrayLastObject( CFArrayRef array ) = CTypeRef
ArrayWriteToURL( CFArrayRef array, CFURLRef url, Boolean atomically ) = Boolean
```

BezierPath

new:

```
BezierPath = BezierPathRef
BezierPathWithOvalInRect( CGRect r ) = BezierPathRef
BezierPathWithRect( CGRect r ) = BezierPathRef
BezierPathWithRoundedRect( CGRect r, CGFloat xRadius, CGFloat yRadius ) = BezierPathRef
BezierPathByFlatteningPath( BezierPathRef path ) = BezierPathRef
BezierPathByReversingPath( BezierPathRef path ) = BezierPathRef
BezierPathMoveToPoint( BezierPathRef path, CGPoint pt )
BezierPathLineToPoint( BezierPathRef path, CGPoint pt )
BezierPathCurveToPoint( BezierPathRef path, CGPoint pt, CGPoint cp1, CGPoint cp2 )
BezierPathClose( BezierPathRef path )
BezierPathRelativeMoveToPoint( BezierPathRef path, CGPoint pt )
BezierPathRelativeLineToPoint( BezierPathRef path, CGPoint pt )
BezierPathRelativeCurveToPoint( BezierPathRef path, CGPoint pt, CGPoint cp1, CGPoint cp2 )
BezierPathAppendPath( BezierPathRef path1, BezierPathRef path2 )
BezierPathAppendPathWithPoints( BezierPathRef path, CGPoint *points, NSInteger count )
BezierPathAppendPathWithOvalInRect( BezierPathRef path, CGRect r )
BezierPathAppendPathWithArcFromPoint( BezierPathRef path, CGPoint pt1, CGPoint pt2, CGFloat radius )
BezierPathAppendPathWithArcWithCenter( BezierPathRef path, CGPoint center, CGFloat radius, CGFloat startAngle, CGFloat endAngle, Boolean clockwise )
BezierPathAppendPathWithRect( BezierPathRef path, CGRect r )
BezierPathAppendPathWithRoundedRect( BezierPathRef path, CGRect r, CGFloat xRadius, CGFloat yRadius )
BezierPathSetWindingRule( BezierPathRef path, NSWindingRule windingRule )
BezierPathSetLineCapStyle( BezierPathRef path, NSLineCapStyle style )
```


BezierPathSetLineJoinStyle(BezierPathRef path, NSLineJoinStyle style)
BezierPathSetLineWidth(BezierPathRef path, CGFloat lineWidth)
BezierPathSetMiterLimit(BezierPathRef path, CGFloat limit)
BezierPathSetFlatness(BezierPathRef path, CGFloat limit)
BezierPathSetLineDash(BezierPathRef path, CGFloat *pattern, NSInteger count, CGFloat phase)
BezierPathSetDefaultWindingRule(NSWindingRule rule)
BezierPathSetDefaultLineCapStyle(NSLineCapStyle style)
BezierPathSetDefaultLineJoinStyle(NSLineJoinStyle style)
BezierPathSetDefaultLineWidth(CGFloat lineWidth)
BezierPathSetDefaultMiterLimit(NSWindingRule limit)
BezierPathSetDefaultFlatness(NSWindingRule flatness)
BezierPathStroke(BezierPathRef path)
BezierPathFill(BezierPathRef path)
BezierPathAddClip(BezierPathRef path)
BezierPathSetClip(BezierPathRef path)
BezierPathClipRect(CGRect r)
BezierPathContainsPoint(BezierPathRef path, CGPoint pt) = Boolean
BezierPathBounds(BezierPathRef path) = CGRect
BezierPathCurrentPoint(BezierPathRef path) = CGPoint
BezierPathsIsEmpty(BezierPathRef path) = Boolean
BezierPathTransformUsingAffineTranform(BezierPathRef path, AffineTransformRef tx)
BezierPathElementCount(BezierPathRef path) = NSInteger
BezierPathElementAtIndex(BezierPathRef path, NSInteger index) = NSBezierPathElement
BezierPathElementAtIndexAssociatedPoints(BezierPathRef path, NSInteger index, CGPoint *points) = NSBezierPathElement
BezierPathRemoveAllPoints(BezierPathRef path)
BezierPathSetAssociatedPointsAtIndex(BezierPathRef path, CGPoint *points, NSInteger index)
BezierPathStrokeFillOvalInRect(CGRect r, CGFloat lineWidth, ColorRef strokeCol, ColorRef fillCol)
BezierPathStrokeFillRect(CGRect r, CGFloat lineWidth, ColorRef strokeCol, ColorRef fillCol)
BezierPathStrokeFillRoundedRect(CGRect r, CGFloat xRadius, CGFloat yRadius, CGFloat lineWidth, ColorRef strokeCol, ColorRef fillCol)
BezierPathStrokeFillPolygon(CFArrayRef pts, CGFloat lineWidth, ColorRef strokeCol, ColorRef fillCol)
BezierPathStrokeOvalInRect(CGRect r, CGFloat lineWidth, ColorRef strokeCol)
BezierPathStrokeRect(CGRect r, CGFloat lineWidth, ColorRef strokeCol)
BezierPathStrokeRoundedRect(CGRect r, CGFloat xRadius, CGFloat yRadius, CGFloat lineWidth, ColorRef strokeCol)
BezierPathStrokeRotatedRect(CGRect r, CGFloat x, CGFloat y, CGFloat angle, CGFloat lineWidth, ColorRef strokeCol)
BezierPathStrokeLine(CGPoint pt1, CGPoint pt2, CGFloat lineWidth, ColorRef strokeCol)
BezierPathStrokeCurve(CGPoint pt1, CGPoint pt2, CGPoint cp1, CGPoint cp2, CGFloat lineWidth, ColorRef strokeCol)
BezierPathStrokePolygon(CFArrayRef pts, CGFloat lineWidth, ColorRef strokeCol)
BezierPathFillOvalInRect(CGRect r, ColorRef fillCol)
BezierPathFillRect(CGRect r, ColorRef fillCol)
BezierPathFillRoundedRect(CGRect r, CGFloat xRadius, CGFloat yRadius, ColorRef fillCol)
BezierPathFillPolygon(CFArrayRef pts, ColorRef fillCol)

Color

new:
ColorBlack = ColorRef
ColorBlue = ColorRef
ColorBrown = ColorRef
ColorClear = ColorRef
ColorCyan = ColorRef
ColorDarkGray = ColorRef
ColorGray = ColorRef
ColorGreen = ColorRef
ColorLightGray = ColorRef
ColorMagenta = ColorRef
ColorOrange = ColorRef
ColorPurple = ColorRef
ColorRed = ColorRef
ColorWhite = ColorRef
ColorYellow = ColorRef

ColorWithCalibratedHSB(CGFloat hue, CGFloat saturation, CGFloat brightness, CGFloat alpha) = ColorRef
ColorWithDeviceHSB(CGFloat hue, CGFloat saturation, CGFloat brightness, CGFloat alpha) = ColorRef
ColorWithHSB(CGFloat hue, CGFloat saturation, CGFloat brightness, CGFloat alpha) = ColorRef// macOS 10.9+
ColorWithSRGB(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha) = ColorRef
ColorWithDisplayP3(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha) = ColorRef// macOS 10.12+
ColorWithRGB(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha) = ColorRef// macOS 10.9+

ColorWithCalibratedRGB(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha) = ColorRef
ColorWithDeviceRGB(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha) = ColorRef
ColorWithWhite(CGFloat white, CGFloat alpha) = ColorRef// macOS 10.9+
ColorWithCalibratedWhite(CGFloat white, CGFloat alpha) = ColorRef
ColorWithDeviceWhite(CGFloat white, CGFloat alpha) = ColorRef
ColorWithGenericGamma22White(CGFloat white, CGFloat alpha) = ColorRef
ColorWithDeviceCMYK(CGFloat cyan, CGFloat magenta, CGFloat yellow, CGFloat black, CGFloat alpha) = ColorRef
ColorWithCGColor(CGColorRef col) = ColorRef// macOS 10.8+
ColorCGColor(ColorRef col) = CGColorRef// macOS 10.8+

ColorSet(ColorRef col)
ColorSetFill(ColorRef col)
ColorSetStroke(ColorRef col)

ColorSetBlack
ColorSetBlue
ColorSetBrown
ColorSetClear
ColorSetCyan
ColorSetDarkGray
ColorSetGray
ColorSetGreen
ColorSetLightGray
ColorSetMagenta
ColorSetOrange
ColorSetPurple
ColorSetRed
ColorSetWhite
ColorSetYellow

ColorSetWithCalibratedHSB(CGFloat hue, CGFloat saturation, CGFloat brightness, CGFloat alpha)
ColorSetWithDeviceHSB(CGFloat hue, CGFloat saturation, CGFloat brightness, CGFloat alpha)
ColorSetWithHSB(CGFloat hue, CGFloat saturation, CGFloat brightness, CGFloat alpha)// macOS 10.9+
ColorSetWithSRGB(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha)
ColorSetWithDisplayP3(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha)// macOS 10.12+
ColorSetWithRGB(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha)// macOS 10.9+
ColorSetWithCalibratedRGB(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha)
ColorSetWithDeviceRGB(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha)
ColorSetWithWhite(CGFloat white, CGFloat alpha)// macOS 10.9+
ColorSetWithCalibratedWhite(CGFloat white, CGFloat alpha)
ColorSetWithDeviceWhite(CGFloat white, CGFloat alpha)
ColorSetWithGenericGamma22White(CGFloat white, CGFloat alpha)
ColorSetWithDeviceCMYK(CGFloat cyan, CGFloat magenta, CGFloat yellow, CGFloat black, CGFloat alpha)
ColorSetWithCGColor(CGColorRef col)// macOS 10.8+

ColorSetFillBlack
ColorSetFillBlue
ColorSetFillBrown
ColorSetFillClear
ColorSetFillCyan
ColorSetFillDarkGray
ColorSetFillGray
ColorSetFillGreen
ColorSetFillLightGray
ColorSetFillMagenta
ColorSetFillOrange
ColorSetFillPurple
ColorSetFillRed
ColorSetFillWhite
ColorSetFillYellow

ColorSetFillWithCalibratedHSB(CGFloat hue, CGFloat saturation, CGFloat brightness, CGFloat alpha)
ColorSetFillWithDeviceHSB(CGFloat hue, CGFloat saturation, CGFloat brightness, CGFloat alpha)
ColorSetFillWithHSB(CGFloat hue, CGFloat saturation, CGFloat brightness, CGFloat alpha)// macOS 10.9+
ColorSetFillWithSRGB(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha)
ColorSetFillWithDisplayP3(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha)// macOS 10.12+
ColorSetFillWithRGB(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha)// macOS 10.9+

ColorSetFillWithCalibratedRGB(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha)
ColorSetFillWithDeviceRGB(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha)
ColorSetFillWithWhite(CGFloat white, CGFloat alpha)// macOS 10.9+
ColorSetFillWithCalibratedWhite(CGFloat white, CGFloat alpha)
ColorSetFillWithDeviceWhite(CGFloat white, CGFloat alpha)
ColorSetFillWithGenericGamma22White(CGFloat white, CGFloat alpha)
ColorSetFillWithDeviceCMYK(CGFloat cyan, CGFloat magenta, CGFloat yellow, CGFloat black, CGFloat alpha)
ColorSetFillWithCGColor(CGColorRef col)// macOS 10.8+

ColorSetStrokeBlack
ColorSetStrokeBlue
ColorSetStrokeBrown
ColorSetStrokeClear
ColorSetStrokeCyan
ColorSetStrokeDarkGray
ColorSetStrokeGray
ColorSetStrokeGreen
ColorSetStrokeLightGray
ColorSetStrokeMagenta
ColorSetStrokeOrange
ColorSetStrokePurple
ColorSetStrokeRed
ColorSetStrokeWhite
ColorSetStrokeYellow

ColorSetStrokeWithCalibratedHSB(CGFloat hue, CGFloat saturation, CGFloat brightness, CGFloat alpha)
ColorSetStrokeWithDeviceHSB(CGFloat hue, CGFloat saturation, CGFloat brightness, CGFloat alpha)
ColorSetStrokeWithHSB(CGFloat hue, CGFloat saturation, CGFloat brightness, CGFloat alpha)// macOS 10.9+
ColorSetStrokeWithSRGB(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha)
ColorSetStrokeWithDisplayP3(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha)// macOS 10.12+
ColorSetStrokeWithRGB(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha)// macOS 10.9+
ColorSetStrokeWithCalibratedRGB(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha)
ColorSetStrokeWithDeviceRGB(CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha)
ColorSetStrokeWithWhite(CGFloat white, CGFloat alpha)// macOS 10.9+
ColorSetStrokeWithCalibratedWhite(CGFloat white, CGFloat alpha)
ColorSetStrokeWithDeviceWhite(CGFloat white, CGFloat alpha)
ColorSetStrokeWithGenericGamma22White(CGFloat white, CGFloat alpha)
ColorSetStrokeWithDeviceCMYK(CGFloat cyan, CGFloat magenta, CGFloat yellow, CGFloat black, CGFloat alpha)
ColorSetStrokeWithCGColor(CGColorRef col)// macOS 10.8+

DialogEvent

new:
DialogEventFont = CTFontRef

Dictionary

new:
DictionaryWithCapacity(NSUInteger numItems) = CFMutableDictionaryRef
DictionaryWithContentsOfURL(CFURLRef url) = CFDictionaryRef
DictionaryWithObject(CTypeRef obj, CFStringRef key) = CFDictionaryRef
DictionaryWithObjects(CTypeRef obj, ...) = CFDictionaryRef
DictionaryWriteToURL(CFDictionaryRef dict, CFURLRef url, Boolean atomically) = Boolean

FileManager

new:
FileManagerURLForTemporaryDirectory = CFURLRef
FileManagerURLForDirectory(NSSearchPathDirectory whichDirectory, NSSearchPathDomainMask domain) = CFURLRef
FileManagerURLForPreferencesDirectory = CFURLRef
FileManagerURLForApplicationDirectory = CFURLRef
FileManagerContentsOfDirectoryAtURL(CFURLRef url, CFArrayRef keys, NSDirectoryEnumerationOptions options) = CFArrayRef
FileManagerCreateDirectoryAtURL(CFURLRef url, Boolean withIntermediateDirectories, CFDictionaryRef attributes) = Boolean
FileManagerCreateFileAtURL(CFURLRef url, CFDataRef contents, CFDictionaryRef attributes) = Boolean
FileManagerRemoveItemAtURL(CFURLRef url) = Boolean
FileManagerReplaceItemAtURL(CFURLRef origURL, CFURLRef newURL) = Boolean
FileManagerTrashItemAtURL(CFURLRef url)
FileManagerCopyItemAtURL(CFURLRef srcURL, CFURLRef dstURL) = Boolean
FileManagerMoveItemAtURL(CFURLRef srcURL, CFURLRef dstURL) = Boolean

FileManagerFileExistsAtURL(CFURLRef url) = Boolean
FileManagerIsReadableFileAtURL(CFURLRef url) = Boolean
FileManagerIsWritableFileAtURL(CFURLRef url) = Boolean
FileManagerIsExecutableFileAtURL(CFURLRef url) = Boolean
FileManagerIsDeletableFileAtURL(CFURLRef url) = Boolean
FileManagerIsDirectoryAtURL(CFURLRef url) = Boolean
FileManagerAttributesOfItemAtURL(CFURLRef url) = CFDictionaryRef
FileManagerSetAttributesOfItemAtURL(CFURLRef url, CFDictionaryRef attributes) = Boolean
FileManagerContentsAtURL(CFURLRef url) = CFDataRef

FontManager

new dialog event:
_fontManagerChangeFont

Graphics

change:
GraphicsCurrentCGContext renamed GraphicsContextCurrentCGContext

Image

new:
ImageNamed(CFStringRef name) = ImageRef
ImageSetName(ImageRef ref, CFStringRef name)
ImageName(ImageRef ref) = CFStringRef
ImageSize(ImageRef ref) = CGSize
ImageSetSize(ImageRef ref, CGSize size)
ImageTypes = CFArrayRef
ImageUnfilteredTypes = CFArrayRef
ImageDrawInRect(ImageRef ref, CGRect rect)
ImageDrawAtPoint(ImageRef ref, CGPoint pt, CGRect fromRect, NSCompositingOperation operation, CGFloat fraction)
ImageBackgroundColor(ImageRef ref) = ColorRef
ImageSetBackgroundColor(ImageRef ref, ColorRef col)
ImageAlignmentRect(ImageRef ref) = CGRect
ImageSetAlignmentRect(ImageRef ref, CGRect rect)
ImageTIFFRepresentation(ImageRef ref) = CFDataRef
ImageCGImageForProposedRect(ImageRef ref, CGRect rect, GraphicsContextRef context, CFDictionaryRef hints) = CGImageRef

ImageView

new:
ImageViewCGImage(NSInteger tag) = CGImageRef

IndexSet

new:
IndexSetWithIndex(NSUInteger value) = IndexSetRef
IndexSetWithIndexesInRange(CFRange range) = IndexSetRef
IndexSetContainsIndex(IndexSetRef set, NSUInteger index) = Boolean
IndexSetContainsIndexes(IndexSetRef set, IndexSetRef subset) = Boolean
IndexSetContainsIndexesInRange(IndexSetRef set, CFRange range) = Boolean
IndexSetIntersectsIndexesInRange(IndexSetRef set, CFRange range) = Boolean
IndexSetCount(IndexSetRef set) = NSUInteger
IndexSetCountOfIndexesInRange(IndexSetRef set, CFRange range) = NSUInteger
IndexSetIsEqual(IndexSetRef set1, IndexSetRef set2) = Boolean
IndexSetFirstIndex(IndexSetRef set) = NSUInteger
IndexSetLastIndex(IndexSetRef set) = NSUInteger

IndexSet() = MutableIndexSetRef
IndexSetAddIndex(MutableIndexSetRef set, NSUInteger index)
IndexSetAddIndexes(MutableIndexSetRef set, IndexSetRef otherSet)
IndexSetAddIndexesInRange(MutableIndexSetRef set, CFRange range)
IndexSetRemoveIndex(MutableIndexSetRef set, NSUInteger index)
IndexSetRemoveIndexes(MutableIndexSetRef set, IndexSetRef otherSet)
IndexSetRemoveAllIndexes(MutableIndexSetRef set)
IndexSetRemoveIndexesInRange(MutableIndexSetRef set, CFRange range)
IndexSetShiftIndexes(MutableIndexSetRef set, NSUInteger index, NSInteger delta)

Number

new:

NumberWithBool(Boolean value) = CFNumberRef
NumberWithChar(char value) = CFNumberRef
NumberWithShort(short value) = CFNumberRef
NumberWithLong(long value) = CFNumberRef
NumberWithLongLong(SInt64 value) = CFNumberRef
NumberWithUnsignedChar(unsigned char value) = CFNumberRef
NumberWithUnsignedShort(unsigned short value) = CFNumberRef
NumberWithUnsignedLong(unsigned long value) = CFNumberRef
NumberWithUnsignedLongLong(UInt64 value) = CFNumberRef
NumberWithFloat(float value) = CFNumberRef
NumberWithDouble(double value) = CFNumberRef

NumberBoolValue(CFNumberRef number) = Boolean
NumberCharValue(CFNumberRef number) = char
NumberShortValue(CFNumberRef number) = short
NumberLongValue(CFNumberRef number) = long
NumberLongLongValue(CFNumberRef number) = SInt64
NumberUnsignedCharValue(CFNumberRef number) = unsigned char
NumberUnsignedShortValue(CFNumberRef number) = unsigned short
NumberUnsignedLongValue(CFNumberRef number) = unsigned long
NumberUnsignedLongLongValue(CFNumberRef number) = UInt64
NumberFloatValue(CFNumberRef number) = float
NumberDoubleValue(CFNumberRef number) = double

NumberStringValue(CFNumberRef number) = CFStringRef
NumberCompare(CFNumberRef number1, CFNumberRef number2) = NSComparisonResult
NumberIsEqualToNumber(CFNumberRef number1, CFNumberRef number2) = Boolean

RunningApplication

new:
RunningApplicationWithProcessIdentifier(pid_t pid) = RunningApplicationRef
RunningApplicationsWithBundleIdentifier(CFStringRef identifier) = CFArrayRef
RunningApplicationCurrentApplication = RunningApplicationRef
RunningApplicationIsActive(RunningApplicationRef ra) = Boolean
RunningApplicationActivate(RunningApplicationRef ra, NSApplicationActivationOptions options) = Boolean
RunningApplicationActivationPolicy(RunningApplicationRef ra) = NSApplicationActivationPolicy
RunningApplicationHide(RunningApplicationRef ra) = Boolean
RunningApplicationUnhide(RunningApplicationRef ra) = Boolean
RunningApplicationIsHidden(RunningApplicationRef ra) = Boolean
RunningApplicationLocalizedString(RunningApplicationRef ra) = CFStringRef
RunningApplicationBundleIdentifier(RunningApplicationRef ra) = CFStringRef
RunningApplicationBundleURL(RunningApplicationRef ra) = CFURLRef
RunningApplicationExecutableArchitecture(RunningApplicationRef ra) = NSInteger
RunningApplicationExecutableURL(RunningApplicationRef ra) = CFURLRef
RunningApplicationLaunchDate(RunningApplicationRef ra) = CFDateRef
RunningApplicationIsFinishedLaunching(RunningApplicationRef ra) = Boolean
RunningApplicationProcessIdentifier(RunningApplicationRef ra) = pid_t
RunningApplicationOwnsMenuBar(RunningApplicationRef ra) = Boolean
RunningApplicationForceTerminate(RunningApplicationRef ra) = Boolean
RunningApplicationTerminate(RunningApplicationRef ra) = Boolean
RunningApplicationIsTerminated(RunningApplicationRef ra) = Boolean
RunningApplicationsTerminateAutomaticallyTerminableApplications

String

new:
StringWithCapacity(NSUInteger capacity) = CFMutableStringRef
StringWithFormat(CFStringRef format, ...) = CFStringRef
StringWithContentsOfURL(CFURLRef url) = CFStringRef
StringWriteToURL(CFStringRef string, CFURLRef url, Boolean atomically, NSStringEncoding encoding) = Boolean
StringSubstringFromIndex(CFStringRef string, CFIndex index) = CFStringRef
StringSubstringWithRange(CFStringRef string, CFRange range) = CFStringRef
StringSubstringToIndex(CFStringRef string, CFIndex index) = CFStringRef
StringByAppendingFormat(CFStringRef string, CFStringRef format, ...) = CFStringRef
StringByAppendingString(CFStringRef string1, CFStringRef string2) = CFStringRef
StringByPaddingToLength(CFStringRef string, NSUInteger newLength, CFStringRef padString, NSUInteger padIndex) = CFStringRef
StringContainsString(CFStringRef string1, CFStringRef string2) = Boolean

StringPropertyList(CFStringRef string) = CTypeRef

Text

new:
TextSelectAll(NSInteger tag)
TextCopy(NSInteger tag)
TextCut(NSInteger tag)
TextPaste(NSInteger tag)
TextCopyFont(NSInteger tag)
TextPasteFont(NSInteger tag)
TextDelete(NSInteger tag)
TextAlignment(NSInteger tag) = NSTextAlignment
TextAlignCenter(NSInteger tag)
TextAlignLeft(NSInteger tag)
TextAlignRight(NSInteger tag)
TextColor(NSInteger tag) = CGColorRef
TextSetColor(NSInteger tag, CGColorRef col)
TextSetColorInRange(NSInteger tag, CGColorRef col, CFRange range)
TextSuperscript(NSInteger tag)
TextSubscript(NSInteger tag)
TextUnscript(NSInteger tag)
TextUnderline(NSInteger tag)
TextSetFont(NSInteger tag, CTFontRef font)
TextSetFontInRange(NSInteger tag, CTFontRef font, CFRange range)
TextReadRTFDFromURL(NSInteger tag, CFURLRef url) = Boolean
TextWriteRTFDToURL(NSInteger tag, CFURLRef url, Boolean atomically) = Boolean

TextView

new dialog events:
_textViewDidChangeSelection
_textViewDoCommandBySelector

URL

new:
URLWithString(CFStringRef string) = CFURLRef
URLFileURLWithPath(CFStringRef string) = CFURLRef
URLPath(CFURLRef url) = CFStringRef
URLLastPathComponent(CFURLRef url) = CFStringRef
URLPathExtension(CFURLRef url) = CFStringRef
URLByAppendingPathComponent(CFURLRef url, CFStringRef pathComponent) = CFURLRef
URLByAppendingPathExtension(CFURLRef url, CFStringRef extension) = CFURLRef
URLByDeletingLastPathComponent(CFURLRef url) = CFURLRef
URLByDeletingPathExtension(CFURLRef url) = CFURLRef
URLResourceValuesForKeys(CFURLRef url, CFArrayRef keys) = CFDictionaryRef

UserDefaults

new:
UserDefaultsObject(CFStringRef key) = CTypeRef
UserDefaultsBool(CFStringRef key) = Boolean
UserDefaultsInteger(CFStringRef key) = long
UserDefaultsDouble(CFStringRef key) = double
UserDefaultsSetObject(CFStringRef key, CTypeRef obj)
UserDefaultsSetBool(CFStringRef key, Boolean value)
UserDefaultsSetDouble(CFStringRef key, double value)
UserDefaultsSetInteger(CFStringRef key, long value)
UserDefaultsRemoveObject(CFStringRef key)
UserDefaultsRegisterDefaults(CFDictionaryRef defaults)
UserDefaultsSynchronize = Boolean
UserDefaultsStoreWindowViewValues(NSInteger wndTag, CFStringRef key)
UserDefaultsRestoreWindowViewValues(NSInteger wndTag, CFStringRef key)
UserDefaultsRemoveWindowViewValues(NSInteger wndTag, CFStringRef key)

Value

new:
ValueWithPointer(ptr p) = ValueRef
ValuePointer(ValueRef v) = ptr
ValueWithRange(CFRange range) = ValueRef

ValueRange(ValueRef v) = CFRange
ValueWithPoint(CGPoint pt) = ValueRef
ValuePoint(ValueRef v) = CGPoint
ValueWithSize(CGSize size) = ValueRef
ValueSize(ValueRef v) = CGSize
ValueWithRect(CGRect r) = ValueRef
ValueRect(ValueRef v) = CGRect

Workspace

new:

WorkspaceOpenURL(CFURLRef url) = Boolean
WorkspaceOpenURLWithApplication(CFURLRef url, CFStringRef appName, Boolean deactivate) = Boolean
WorkspaceOpenURLsWithApplicationAtURL(CFArrayRef urls, CFURLRef appURL, NSWorkspaceLaunchOptions options, CFDictionaryRef configuration) = Boolean
WorkspaceOpenURLWithApplicationAtURL(CFURLRef url, CFURLRef appURL, NSWorkspaceLaunchOptions options, CFDictionaryRef configuration) = Boolean
WorkspaceLaunchApplication(CFStringRef appName) = Boolean
WorkspaceHideOtherApplications
WorkspaceDuplicateURLs(CFArrayRef urls)
WorkspaceRecycleURL(CFURLRef url)
WorkspaceRecycleURLs(CFArrayRef urls)
WorkspaceTypeOfFileAtURL(CFURLRef url) = CFStringRef
WorkspacePreferredFilenameExtensionForType(CFStringRef typeName) = CFStringRef
WorkspaceFilenameExtensionIsValidForType(CFStringRef filenameExtension, CFStringRef typeName) = Boolean
WorkspaceFileTypeConformToType(CFStringRef firstTypeName, CFStringRef secondTypeName) = Boolean
WorkspaceGetInfoForFileAtURL(CFURLRef url, CFStringRef *appName, CFStringRef *type) = Boolean
WorkspaceURLForApplicationToOpenURL(CFURLRef url) = CFURLRef
WorkspacesFilePackageAtURL(CFURLRef url) = Boolean
WorkspaceFrontmostApplication = RunningApplicationRef
WorkspaceRunningApplications = CFArrayRef
WorkspaceURLForApplicationWithIdentifier(CFStringRef identifier) = CFURLRef
WorkspaceNoteFileSystemChangedAtURL(CFURLRef url)

20171202 FB 5.7.111

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.110]

FB 5.7.111 requires OS X 10.6 or higher

[01] OSPanelSave accessory view example using CocoaUI added to Examples/Files/OSPanel examples
[02] New CocoaUI demos: TextField Arrow Keys, TableView Sort, MiniwindowImage, FontPanel, AboutPanelWithOptions

CocoaUI changes

General

The imageName param of the following functions and statements now optionally accepts a path to a resource image.

Functions:

AlertsetIcon
ButtonsetImage
ButtonSeAlternatetImage
CursorsetImage
LevelIndicatorSetRatingPlaceholderImage
LevelIndicatorsetImage
SegmentedControlSetSegment

Statements:

imageview
toolbaritem

Application

new:

AppOrderFrontStandardAboutPanel


```
AppOrderFrontStandardAboutPanelWithOptions( CFDictionaryRef options )    // see
AboutPanelWithOptions demo
AppReplyToApplicationShouldTerminate( Boolean flag )
```

DialogEvent

```
new:
_controlDoCommandBySelector    // see TextField Arrow Keys demo
DialogEventString = CFStringRef
```

FontManager // see FontPanel demo

```
new:
FontManagerSetSelectedFont( CTFontRef font )
FontManagerSelectedFont = CTFontRef
FontManagerShowPanel
FontManagerClosePanel
```

MenuItem

```
new:
MenuItemSetOnMenuAction( SInt32 menuIndex, SInt32 itemIndex )
```

SearchField

```
new:
SearchFieldRecentSearches( SInt32 tag ) = CFArrayRef
```

SegmentedControl

```
new:
SegmentedControlSelectSegmentWithTag( SInt32 ctrlTag, SInt32 segTag ) = Boolean
```

ScrollView

```
new:
ScrollViewContentSize( SInt32 tag ) = CGSize
ScrollViewDocumentVisibleRect( SInt32 tag ) = CGRect
ScrollViewBackgroundColor( SInt32 tag ) = CGColorRef
ScrollViewSetBackgroundColor( SInt32 tag, CGColorRef col )
ScrollViewSetBackgroundRGB( SInt32 tag, CGFloat red, CGFloat green, CGFloat blue, CGFloat alpha
)
ScrollViewDrawsBackground( SInt32 tag ) = Boolean
ScrollViewSetDrawsBackground( SInt32 tag, Boolean flag )
ScrollViewBorderType( SInt32 tag ) = NSBorderType
ScrollViewSetBorderType( SInt32 tag, NSBorderType type )
ScrollViewDocumentCursor( SInt32 tag ) = CursorRef
ScrollViewSetDocumentCursor( SInt32 tag, CursorRef ref )
ScrollViewHasHorizontalRuler( SInt32 tag ) = Boolean
ScrollViewSetHasHorizontalRuler( SInt32 tag, Boolean flag )
ScrollViewHasVerticalRuler( SInt32 tag ) = Boolean
ScrollViewSetHasVerticalRuler( SInt32 tag, Boolean flag )
ScrollViewRulersVisible( SInt32 tag ) = Boolean
ScrollViewSetRulersVisible( SInt32 tag, Boolean flag )
ScrollViewLineScroll( SInt32 tag ) = CGFloat
ScrollViewSetLineScroll( SInt32 tag, CGFloat value )
ScrollViewHorizontalLineScroll( SInt32 tag ) = CGFloat
ScrollViewSetLHorizontalLineScroll( SInt32 tag, CGFloat value )
ScrollViewVerticalLineScroll( SInt32 tag ) = CGFloat
ScrollViewSetVerticalLineScroll( SInt32 tag, CGFloat value )
ScrollViewPageScroll( SInt32 tag ) = CGFloat
ScrollViewSetPageScroll( SInt32 tag, CGFloat value )
ScrollViewHorizontalPageScroll( SInt32 tag ) = CGFloat
ScrollViewSetHorizontalPageScroll( SInt32 tag, CGFloat value )
ScrollViewVerticalPageScroll( SInt32 tag ) = CGFloat
ScrollViewSetVerticalPageScroll( SInt32 tag, CGFloat value )
ScrollViewScrollsDynamically( SInt32 tag ) = Boolean
ScrollViewSetScrollsDynamically( SInt32 tag, Boolean flag )
ScrollViewAllowsMagnification( SInt32 tag ) = Boolean// macOS 10.8+
ScrollViewSetAllowsMagnification( SInt32 tag, Boolean flag )// macOS 10.8+
ScrollViewMagnification( SInt32 tag ) = CGFloat// macOS 10.8+
ScrollViewSetMagnification( SInt32 tag, CGFloat value )// macOS 10.8+
ScrollViewMagnifyToFitRect( SInt32 tag, CGRect r )// macOS 10.8+
```

```
ScrollViewMaxMagnification( SInt32 tag ) = CGFloat// macOS 10.8+
ScrollViewSetMaxMagnification( SInt32 tag, CGFloat value )// macOS 10.8+
ScrollViewMinMagnification( SInt32 tag ) = CGFloat// macOS 10.8+
ScrollViewSetMinMagnification( SInt32 tag, CGFloat value )// macOS 10.8+
ScrollViewSetMagnificationCenteredAtPoint( SInt32 tag, CGFloat value, CGPoint pt )// macOS
10.8+
```

Slider

```
new:
SliderType( SInt32 tag ) = NSSliderType
SliderAltIncrementValue( SInt32 tag ) = double
SliderSetAltIncrementValue( SInt32 tag, double value )
```

TableView

```
new:
Column sorting    // see TableView Sort demo
```

changes:

TableViewSetData now retains the array

TextField

```
new:
TextFieldSetSelectedRange( SInt32 tag, CFRange range )
TextFieldSelectedRange( SInt32 tag ) = CFRange
```

Window

```
new:
WindowIsKey( SInt32 tag ) = Boolean
WindowCanBecomeKey( SInt32 tag ) = Boolean
WindowMakeKey( SInt32 tag )
WindowMakeKeyAndOrderFront( SInt32 tag )
WindowIsMain( SInt32 tag ) = Boolean
WindowCanBecomeMain( SInt32 tag ) = Boolean
WindowMakeMain( SInt32 tag )
WindowPerformClose( SInt32 tag )
WindowIsMiniaturized( SInt32 tag ) = Boolean
WindowPerformMiniaturize( SInt32 tag )
WindowMiniaturize( SInt32 tag )
WindowDeminiaturize( SInt32 tag )
WindowSetMiniwindowImage( SInt32 tag, CFStringRef imageName )    // see MiniwindowImage demo
WindowSetMiniwindowTitle( SInt32 tag, CFStringRef title )
WindowPrint( SInt32 tag )
WindowContentBorderThickness( SInt32 tag, CGRectEdge edge ) = CGFloat
WindowSetPreventsApplicationTerminationWhenModal( SInt32 tag, Boolean flag )
WindowIsSheet( SInt32 tag ) = Boolean
WindowOrderOut( SInt32 tag )
WindowOrderFront( SInt32 tag )
WindowOrderFrontRegardless( SInt32 tag )
WindowSaveFrameUsingName( SInt32 tag, CFStringRef name )
```

20171202 FB 5.7.110

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.109]

FB 5.7.110 requires OS X 10.6 or higher

[01] FB's I/O commands (like LINE INPUT) would only read or write a partial file and not all of it. This was caused by inappropriate accumulation of file

handles (NSFileHandle) which resulted in exhaustion of available file descriptors.

Typically this only occurred when FB file I/O was transferring one byte at a time. Fixed (I hope).

20171130 FB 5.7.109

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.108]

FB 5.7.109 requires OS X 10.6 or higher

[01] FB's EOF() function failed.

[02] TitlebarAccessoryViewWithTag() causes crash when attempting titlebar access for a window without one.

20171120 FB 5.7.108

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.107]

FB 5.7.108 requires OS X 10.6 or higher

[01] Lots of CocoaUI additions and changes - see "CocoaUI changes" below for details

[02] New CocoaUI demos for

GCD dispatch

TextView Attributed strings

TitlebarAccessoryView

DragDrop View

Several CocoaUI demos received bug fixes and updates to reflect current CocoaUI syntax.

[03] FB I/O keyword code (i.e. OPEN, READ/WRITE, LOF, RECORD, APPEND etc.) updated

It should deliver more consistent cooperative advisory locking on local storage. Network advisory locking not supported.

Watch for a separate list post.

[04] FBtoC crashes instead of reporting a Bad Factor on 'end fn = r' where 'r' is an undefined record name.

[05] FBtoC's Make Xcode Project(with stripping) now works.

[06] A unix function changed its parameter order. FBtoC code adjusted to conform.

CocoaUI changes

Application

new:

AppSetOutputWindow(SInt32 tag)

Box

new:

BoxContentViewMargins(SInt32 tag) = CGSize

BoxSetContentViewMargins(SInt32 tag, CGSize size)

ColorWell

new:

ColorWellSetColor(SInt32 tag, CGColorRef col)

Control

new:

ControlSetFormat(SInt32 tag, CFStringRef characters, Boolean pass, UInt32 maxLength, Boolean caseSensitive)

DialogEvent

new:

DialogEventArray = CFArrayRef

Menu

new:

MenuItemSetAlternate(SInt32 menuIndex, SInt32 itemIndex, Boolean flag)

MenuAddItem(SInt32 menuIndex, CFStringRef title, CFStringRef keyEquivalent, NSEventModifierFlags keyEquivalentModifierMask)

MenuInsertItem(SInt32 menuIndex, SInt32 itemIndex, CFStringRef title, CFStringRef keyEquivalent, NSEventModifierFlags keyEquivalentModifierMask)

SplitView

fix:

Unable to modify embedded views in nib splitview.

new:

standard menu statements

filemenu, findmenu, formatmenu, fontmenu, textmenu, viewmenu, helpmenu

TableView

new:

TableViewSelectRow(SInt32 tag, SInt32 rowIndex)

Text

new:

TextSetString(SInt32 tag, CFStringRef string)

TextSelectedRange(SInt32 tag) = CFRange

TextSetSelectedRange(SInt32 tag, CFRange range)

TextField

new:

TextFieldSetAllowsUndo(SInt32 tag, Boolean flag)

changes:

TextFieldSetFormat removed - use ControlSetFormat instead.

fix:

Focus ring appeared in wrong place when field's parent view flipped.

TextView

new:

TextViewSetAllowsUndo(SInt32 tag, Boolean flag)

TextViewSetImportsGraphics(SInt32 tag, Boolean flag)

TextViewSetAllowsImageEditing(SInt32 tag, Boolean flag)

TextViewSetAutomaticLinkDetectionEnabled(SInt32 tag, Boolean flag)

TextViewToggleAutomaticLinkDetection(SInt32 tag)

TextViewSetDisplaysLinkToolTips(SInt32 tag, Boolean flag)

TextViewSetAutomaticQuoteSubstitutionEnabled(SInt32 tag, Boolean flag)

TextViewToggleAutomaticQuoteSubstitution(SInt32 tag)

changes:

Default AllowsUndo setting is now `_true`.

TextViewString and TextViewSetString removed - use TextString and TextSetString instead.

View

new:

ViewAutoresizesSubviews(SInt32 tag) = Boolean

ViewSetAutoresizesSubviews(SInt32 tag, Boolean flag)

ViewRegisterForDraggedFiles(SInt32 tag)

dialog constants

`_viewDraggingEntered`

`_viewDraggingExited`

`_viewPerformDragOperation`

Window

new:

WindowViewWithTag(SInt32 wndTag, SInt32 viewTag) = ptr

WindowSetAutorecalculatesKeyViewLoop(SInt32 tag, Boolean flag)

WindowRecalculateKeyViewLoop(SInt32 tag)

WindowRecalculateKeyViewLoopByTagOrder(SInt32 tag)

WindowFirstResponder(SInt32 tag) = SInt32

WindowInitialFirstResponder(SInt32 tag) = SInt32

WindowSetInitialFirstResponder(SInt32 wndTag, SInt32 responderTag)

WindowContentRectForFrameRect(SInt32 tag, CGRect r) = CGRect

WindowFrameRectForContentRect(SInt32 tag, CGRect r) = CGRect

WindowSheetParent(SInt32 tag) = SInt32// macOS 10.9+

WindowSetResizeIncrements(SInt32 tag, CGSize size)

```
WindowSetContentSize( SInt32 tag, CGSize size )
WindowSetContentSizeIncrements( SInt32 tag, CGSize size )
WindowOrderRelativeTo( SInt32 wndTag, NSWindowOrderingMode order, SInt32 otherWndTag )
WindowSetLevel( SInt32 tag, NSWindowLevel level )
WindowStandardWindowButton( SInt32 tag, NSWindowButton btn ) = ptr
WindowSetTitleVisibility( SInt32 tag, NSWindowTitleVisibility visibility )
WindowResizeFlags( SInt32 tag ) = NSEventModifierFlags
```


20171030 FB 5.7.107

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.106]

FB 5.7.107 requires OS X 10.6 or higher

General Fixes & Additions

- [01] Util_Array.incl's ArrayFirstObject() now works with 10.6+
- [02] Editor crash opening a file when using _kFBProjTypeAlt1 = _"FB^π" because this constant translates to zero in C.
- [03] New 'Abstract' demo and ViewSetNeedsDisplay demo added to CocoaUI Demos/Graphics.

- [04] A later(Brian 20171101) supplement to 5.7.107 fixed a bad structure bug that causes FBtoC to quit. Issue happens when user fails to complete a block (like 'end select'). Code fix by Michelle.

CocoaUI changes 20171030(addendum to .107 by adding to .107's bundle)

Box

new:

```
BoxSetBorderRGB( SInt32 tag, CGFloat r, CGFloat g, CGFloat b, CGFloat a )
BoxSetFillRGB( SInt32 tag, CGFloat r, CGFloat g, CGFloat b, CGFloat a )
BoxSetTitleFontWithName( SInt32 tag, CFStringRef fontName, CGFloat size )
```

Button

new:

```
ButtonSetTitleRGB( SInt32 tag, CGFloat r, CGFloat g, CGFloat b, CGFloat a )
ButtonSetBackgroundRGB( SInt32 tag, CGFloat r, CGFloat g, CGFloat b, CGFloat a )
```

CocoaAlert

changes:

Previously, a sheet alert's tag value and return code would appear in dialog(-1), dialog(ev) respectively. They're now the other way around: returnCode = dialog(ev), alertTag = dialog(-1).

ColorPanel

new:

```
ColorPanelSetRGB( CGFloat r, CGFloat g, CGFloat b, CGFloat a )
```

ColorWell

new:

```
ColorWellSetRGB( SInt32 tag, CGFloat r, CGFloat g, CGFloat b, CGFloat a )
```

Control

new:

dialog event constants - these are same values as _textFieldDidBeginEditing, etc

```
_controlTextDidBeginEditing
_controlTextDidChange
_controlTextDidEndEditing
```

```
ControlSetDoubleValue( SInt32 tag, double value )
ControlSetIntegerValue( SInt32 tag, SInt32 value )
ControlSetStringValue( SInt32 tag, CFStringRef string )
```

changes:

ControlSetActionCallback. The order of the callback function parameters has changed:

was: MyControlCallback(ctrlTag, wndTag, event)
now: MyControlCallback(event, ctrlTag, wndTag)

DatePicker

new:
DatePickerSetTextRGB(SInt32 tag, CGFloat r, CGFloat g, CGFloat b, CGFloat a)
DatePickerSetBackgroundRGB(SInt32 tag, CGFloat r, CGFloat g, CGFloat b, CGFloat a)

Menu

new:
MenuNumberOfItems(SInt32 menuIndex) = SInt32

PopUpButton

new:
PopUpButtonSetTitle(SInt32 tag, CFStringRef title)
PopUpButtonSetPreferredEdge(SInt32 tag, CGRectEdge edge)
PopUpButtonSetAutoenablesItems(SInt32 tag, Boolean flag)
PopUpButtonItemSetEnabled(SInt32 tag, SInt32 index, Boolean flag)

TextField

new:
TextFieldSetTextRGB(SInt32 tag, CGFloat r, CGFloat g, CGFloat b, CGFloat a)
TextFieldSetBackgroundRGB(SInt32 tag, CGFloat r, CGFloat g, CGFloat b, CGFloat a)

TextView

new:
TextViewSetBackgroundRGB(SInt32 tag, CGFloat r, CGFloat g, CGFloat b, CGFloat a)

View

new:
ViewScrollRect(SInt32 tag, CGRect r, CGSize size)
ViewSetNeedsDisplayInRect(SInt32 tag, CGRect r)
ViewFrameRotation(SInt32 tag) = CGFloat
ViewBoundsRotation(SInt32 tag) = CGFloat
ViewsEmbedInView(SInt32 tag, ...)
ViewSetAcceptsFirstResponder(SInt32 tag, Boolean flag)

dialog events:
_viewKeyDown
_viewKeyUp
_viewFlagsChanged

Window

new:
WindowSetBackgroundRGB(SInt32 tag, CGFloat r, CGFloat g, CGFloat b, CGFloat a)
WindowSubclassContentView(SInt32 tag)

CocoaUI Changes/Additions(for .107)

Application

new:
AppSetTimer(CTimeInterval interval, ptr callback, Boolean repeats)

Button

new:
ButtonSetImageScaling(SInt32 tag, NSImageScaling scaling)
ButtonSetBackgroundColor(SInt32 tag, CGColorRef)// macOS 10.8+
ButtonSetImageDimsWhenDisabled(SInt32 tag, Boolean flag)

change:
A new button given a title of "OK" (caps) is made the default (blue) button and responds to the return key.
A new button given a title of "Cancel" (case-sensitive) is made the cancel button and responds to the esc key and cmd-<full stop>.

CocoaAlert

new function and statement:
cocoaalert(tag, style, msg, info, buttonTitles) = SInt32 // currently requires Cocoa runtime (CocoaInit)
cocoaalert tag, style, msg, info, buttonTitles, sheetFlag

Control

new:
ControlSetAction(SInt32 tag, CFStringRef actionName)
ControlSetActionCallback(SInt32 tag, ptr callback)
ControlSetEnabled(SInt32 tag, Boolean flag)

changes:
ControlSetFontWithName - if name is null or zero-length, previous font is used. If size == 0, previous size is used.

DatePicker

new:
DatePickerSetBezeled(SInt32 tag, Boolean flag)
DatePickerSetBordered(SInt32 tag, Boolean flag)

Menu

new:
windowmenu statement - puts up default Window menu in menu bar.

MenuItemSetAction(SInt32 menuIndex, SInt32 itemIndex, CFStringRef actionName)
MenuBarSetVisible(Boolean flag)
MenuBarHeight = CGFloat
MenuRemoveItem(SInt32 menuIndex, SInt32 itemIndex)
MenuRemoveAllItems(SInt32 menuIndex)
MenuItemWithTitle(SInt32 menuIndex, CFStringRef title) = SInt32
MenuPopUp(SInt32 menuIndex, SInt32 itemIndex, CGPoint location, SInt32 viewTag) = Boolean
MenuSetShowsStateColumn(SInt32 menuIndex, Boolean flag)
MenuSetFont(SInt32 menuIndex, CTFontRef font)
MenuSetFontWithName(SInt32 menuIndex, CFStringRef name, CGFloat size)
MenuItemSetState(SInt32 menuIndex, SInt32 itemIndex, NSControlStateValue state)
MenuSetValidateItemsCallback(ptr callback)

changes:
The cocoa menu statement no longer needs '-1' item index to refer to the menu itself - simply omit the item index param from the statement. e.g. cocoa menu 1,,, @"File"
Menus with index value > 100 are not added to the menu bar but stored elsewhere for later use (submenus, etc).

fix:
MenuSetSubmenu now works

NibMenu

change:
A menu assigned a tag value > 100 is given that same value to its index (see Menu changes above)

Panel

change:
options param removed from statement. Options are now set by function calls (see below)

new:
PanelSetFloatingPanel(SInt32 tag, Boolean flag)
PanelSetBecomesKeyOnlyIfNeeded(SInt32 tag, Boolean flag)
PanelSetWorksWhenModal(SInt32 tag, Boolean flag)

Popover

new:
dialog events
_popoverDetachableWindow
_popoverShouldClose
_popoverWillShow
_popoverDidShow
_popoverWillClose
_popoverDidClose
_popoverDidDetach// macOS 10.10
_popoverShouldDetach// macOS 10.10

PopoverSetDetachableWindow(SInt32 popoverTag, SInt32 wndTag)

PopUpButton

new:
PopUpButonSetMenu(SInt32 tag, SInt32 menuIndex)

ProgressIndicator

change:
style param removed from statement. Style is now set by function calls (see below)

new:
ProgressIndicatorSetStyle(SInt32 tag, UInt32 style)

ProgressIndicatorSetIndeterminate(SInt32 tag, Boolean flag)

ScrollView

change:
options paramenter removed from statement. Options are now set by functions (see below)

new:
ScrollViewSetHasHorizontalScroller(SInt32 tag, Boolean flag)
ScrollViewSetHasVerticalScroller(SInt32 tag, Boolean flag)
ScrollViewSetAutohidesScrollers(SInt32 tag, Boolean flag)
ScrollViewFlashScrollers(SInt32 tag)

SearchField

change:
options parameter removed. Options are now set by functions (see below)

new:
SearchFieldSetSendsSearchStringImmediately(SInt32 tag, Boolean flag)// macOS 10.10
SearchFieldSetSendsWholeSearchString(SInt32 tag, Boolean flag)// macOS 10.10
SearchFieldSetCentersPlaceholder(SInt32 tag, Boolean flag)// macOS 10.11
SearchFieldSetMaximumRecents(SInt32 tag, SInt32 value)// macOS 10.10
SearchFieldSetRecentsAutosaveName(SInt32 tag, CFStringRef name)
SearchFieldSetSearchMenuTemplate(SInt32 tag, SInt32 menuIndex)// macOS 10.10

SegmentedControl

change:
style and mode parameters removed from statement. Style and mode are now set by functions (see below)

new:
SegmentedControlSetMenu(SInt32 tag, SInt32 segmentIndex, SInt32 menuIndex)
SegmentedControlSetStyle(SInt32 tag, NSSegmentStyle style)
SegmentedControlSetTrackingMode(SInt32 tag, NSSegmentSwitchTracking mode)

Slider

change:
style parameter removed from statement. Style now set by functions (see below + ControlSetContinuous)

new:
SliderSetType(SInt32 tag, NSSliderType type)
SliderSetAllowsTickMarkValuesOnly(SInt32 tag, Boolean flag)
SliderSetTickMarkPosition(SInt32 tag, NSTickMarkPosition position)

Stepper

change:
valueWraps value defaults to _false

TableView

new:
now issues dialog event _btnClick
_tableViewDoubleAction dialog event
_tableViewSelectionDidChange dialog event

TableViewData(SInt32 tag) = CFMutableArrayRef
TableViewReloadData(SInt32 tag)
TableViewScrollRowToVisible(SInt32 tag, SInt32 row)
TableViewScrollColumnToVisible(SInt32 tag, SInt32 col)
TableViewSelectedRowIndexes(SInt32 tag) = CFArrayRef
TableViewNumberOfSelectedRows(SInt32 tag) = SInt32
TableViewIsRowSelected(SInt32 tag, SInt32 row) = Boolean
TableViewClickedRow(SInt32 tag) = SInt32
TableViewClickedColumn(SInt32 tag) = SInt32

changes:
Now allows unlimited number of columns
Editable columns supported
TableViewAddRow function removed

TextField

new:
TextFieldSetAllowsEditingTextAttributes(SInt32 tag, Boolean flag)
TextFieldSetImportsGraphics(SInt32 tag, Boolean flag)

changes:
TextFieldStringValue function removed - use ControlStringValue instead

TextView

new:
TextViewPerformFindPanelAction(SInt32 tag, NSTextFinderAction action)

Toolbar

change:
displayMode and options parameters removed from statement. These are now set by functions (see below)

new:
ToolbarSetDisplayMode(SInt32 tag, NSToolbarDisplayMode mode)
ToolbarSetAllowsUserCustomization(SInt32 tag, Boolean flag)
ToolbarSetShowsBaselineSeparator(SInt32 tag, Boolean flag)
ToolbarSetAutosavesConfiguration(SInt32 tag, Boolean flag)
ToolbarSetAllowsExtensionItems(SInt32 tag, Boolean flag)

ToolbarItem

new:
ToolbarItemSetToolTip(SInt32 toolbarTag, SInt32 itemTag, CFStringRef toolTip)

View

new:
ViewIsHidden(SInt32 tag) = Boolean
ViewIsHiddenOrHasHiddenAncestor(SInt32 tag) = Boolean
ViewNextKeyView(SInt32 tag) = SInt32
ViewNextValidKeyView(SInt32 tag) = SInt32
ViewPreviousKeyView(SInt32 tag) = SInt32
ViewPreviousValidKeyView(SInt32 tag) = SInt32

Window

new:
WindowSetDelegateCallback(SInt32 tag, ptr callback)
WindowAddChildWindow(SInt32 parTag, SInt32 childTag, NSWindowOrderingMode ordered)
WindowRemoveChildWindow(SInt32 parTag, SInt32 childTag)
WindowParentWindow(SInt32 tag) = SInt32
WindowSetHidesOnDeactivate(SInt32 tag, Boolean flag)
WindowSetHasShadow(SInt32 tag, Boolean flag)
WindowSetFrameOrigin(SInt32 tag, CGPoint pt)
WindowSetFrame(SInt32 tag, CGRect frame)
WindowCascadeTopLeftFromPoint(SInt32 tag, CGPoint pt)
WindowIsZoomed(SInt32 tag) = Boolean
WindowPerformZoom(SInt32 tag)
WindowZoom(SInt32 tag)
WindowSetShowsResizeIndicator(SInt32 tag, Boolean flag)
WindowSetMaxFullScreenContentSize(SInt32 tag, CGSize size)// macOS 10.11+
WindowSetMinFullScreenContentSize(SInt32 tag, CGSize size)// macOS 10.11+
WindowOrderBack(SInt32 tag)
WindowIsVisible(SInt32 tag) = Boolean
WindowRemoveFrameUsingName(CFStringRef name)
WindowSetExcludedFromWindowsMenu(SInt32 tag, Boolean flag)

20171005 FB 5.7.106

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.105]

FB 5.7.106 requires OS X 10.6 or higher

General Fixes & Additions

- [01] Programmatically creating toolbars with Source-file combining set to OFF had a bug.
- [02] TWM (The Window Maker) updated to version 1.9.15
- [03] Apple macro name changed causing crash in FileHandling.c (__Require_noErr is new name)
- [04] CFIndex.incl header and its runtime file index.c now 64-bit compatible
- [05] New Util_Sound.incl header and examples.

[06] Get/Put/Kill Preferences statements now are 64-bit.
Note: this work did NOT involve the headers Util_CFPrefs.incl and Util_CFPrefsCFStrings.incl.
[07] Util_Array.incl updated with two new convenience accessors, ArrayFirstObject() and ArrayLastObject()
[08] Quartz Cocoa.bas demo (for Quartz 1.15 in FB Examples/Graphics/CoreGraphics(Quartz)) updated to use GraphicsCurrentCGContext and ViewBounds().
[09] FB's file I/O OPEN statement does *not* honor POSIX advisory locking when data files are located on a server (either macOS Server or macOS File Sharing). This means OPEN "N" and OPEN "R" will not return appropriate errors if the file is already locked by another process. Paolo C. suggested a runtime change which works in his environment (see "File Bug in 5.7.105" list thread in late September 2017 for details) but it isn't a general solution. FBers needing to access server files should consider other file access methods other than OPEN if their files are on servers.

CocoaUI Changes/Additions

Application

New:

_appOpenURLs event constant

AppEventString = CFStringRef
AppEventArray = CFArrayRef
AppEventDictionary = CFDictionaryRef
AppEventURL = CFURLRef

Changes:

_appOpenFile and _appOpenFiles constants removed. Use _appOpenURLs instead.

ColorPanel

New:

ColorPanelColor = CGColorRef
ColorPanelSetColor(CGColorRef col)
ColorPanelSetMode(NSColorPanelMode mode)
ColorPanelSetShowsAlpha(Boolean flag)
ColorPanelAlpha = CGFloat
ColorPanelShow
ColorPanelClose

Control

New:

ControlStringValue(SInt32 tag) = CFStringRef
ControlTakeDoubleValueFrom(SInt32 tag, SInt32 fromViewTag)
ControlTakeIntegerValueFrom(SInt32 tag, SInt32 fromViewTag)
ControlIsEnabled(SInt32 tag) = Boolean
ControlSizeThatFits(SInt32 tag, CGSize size) = CGSize // macOS 10.10
ControlSetContinuous(SInt32 tag, Boolean flag)
ControlPerformClick(SInt32 tag)
ControlSetLineBreakMode(SInt32 tag, NSLineBreakMode mode)

Dialog

Changes:

DialogEventCGContext removed. Use GraphicsCurrentCGContext instead.

DialogEventRect no longer returns the view rect, and now returns its dirtyRect. To get the view rect, use either ViewFrame or ViewBounds, whichever is appropriate.

Event

New:

EventLocationInWindow = CGPoint
EventModifierFlags = NSEventModifierFlags
EventTimestamp = CFTimeInterval
EventType = NSEventType
EventWindow = ptr
EventKeyRepeatDelay = CFTimeInterval
EventKeyRepeatInterval = CFTimeInterval

EventCharacters = CFStringRef
EventCharactersIgnoringModifiers = CFStringRef
EventIsARepeat = Boolean
EventKeyCode = unsigned short
EventPressedMouseButtons = UInt32
EventDoubleClickInterval = CFTimeInterval
EventMouseLocation = CGPoint
EventClickCount = SInt32
EventLocationInView(SInt32 tag) = CGPoint

Geometry
Geometry.incl header removed. That header contained NSRectEdge constants NSMinXEdge, NSMinYEdge, NSMaxXEdge & NSMaxYEdge. Use: CGRectXxxx. instead.

Graphics
New:
GraphicsCurrentCGContext = CGContextRef

ImageView
Syntax change:
options param removed. Attributes are now set with functions (see below).

New:
ImageViewSetAnimates(SInt32 tag, Boolean flag)
ImageViewSetEditable(SInt32 tag, Boolean flag)
ImageViewSetAllowsCutCopyPaste(SInt32 tag, Boolean flag)

LevelIndicator
Syntax change:
options param removed. Attributes are now set with functions (see below).

New:
LevelIndicatorSetEditable(SInt32 tag, Boolean flag)
LevelIndicatorSetTickMarkPosition(SInt32 tag, NSTickMarkPosition position)

Menu
Change:
If tag values are given to menus and items, those values are now returned to FB's on menu function instead of their index values.

New:
MenuSetTag(SInt32 menuIndex, SInt32 itemIndex, SInt32 tag)
MenuSetAutoenablesItems(SInt32 menuIndex, Boolean flag)

SegmentedControl
Fixed: Crash when setting segment image. This crash only occurred on machines running macOS 10.12 or earlier and when multiple segmented controls in a window.

Stepper
Syntax change:
options param removed. These attributes are now set with functions (see below).

StepperSetAutorepeat(SInt32 tag, Boolean flag)
StepperSetValueWraps(SInt32 tag, Boolean flag)
StepperSetContinuous(SInt32 tag, Boolean flag)

TextField
Syntax changes:
behavior param removed. These attributes are now set with functions (see below).
lineBreakMode param removed. This attribute is now set with a function (see Control.incl).

New:
TextFieldSetEditable(SInt32 tag, Boolean flag)
TextFieldSetSelectable(SInt32 tag, Boolean flag)

TextLabel
Syntax change:
lineBreakMode param removed. This attribute is now set with a function (see Control.incl).

View
Syntax change:

subclass param added (default = `_false`). Enabling subclass allows the user to pick up `_viewDrawRect` plus other dialog events (see below).

```
New:
ViewAddSubviewPositioned( SInt32 superviewTag, SInt32 subviewTag, NSWindowOrderingMode position, SInt32 otherViewTag )
ViewSetWantsLayer( SInt32 tag, Boolean flag )
ViewSetFrame( SInt32 tag, CGRect r )
ViewSetFlipped( SInt32 tag, Boolean flag )
```

```
ViewAnimator( SInt32 tag ) = ptr
ViewAnimatorSetFrame( SInt32 tag, CGRect r )
ViewAnimatorSetFrameOrigin( SInt32 tag, CGPoint origin )
ViewAnimatorSetFrameSize( SInt32 tag, CGSize size )
ViewAnimatorSetFrameRotation( SInt32 tag, CGFloat rot )
ViewAnimatorSetBounds( SInt32 tag, CGRect r )
ViewAnimatorSetBoundsOrigin( SInt32 tag, CGPoint origin )
ViewAnimatorSetBoundsSize( SInt32 tag, CGSize size )
ViewAnimatorSetBoundsRotation( SInt32 tag, CGFloat rot )
ViewAnimatorSetFrameCenterRotation( SInt32 tag, CGFloat rot )
ViewAnimatorSetAlphaValue( SInt32 tag, CGFloat value )
```

```
dialog events:
_viewMouseDown
_viewMouseDragged
_viewMouseUp
_viewMouseMove
_viewMouseEntered
_viewMouseExited
_viewRightMouseDown
_viewRightMouseDragged
_viewRightMouseUp
```

Window

```
New:
_windowContentViewTag - this constant identifies a window's content view
```

```
WindowSetTitle( SInt32 tag, CFStringRef title ) // this now works for both Carbon and Cocoa windows
```

Demos:
Couple of new graphic demos showing `GraphicsCurrentCGContext()` and `ViewBounds()` usage

FB Runtime Status

64-bit Runtime	Comments
AppThings.m	
CocoaUI.m	
Containers.c	
FileHandling.c	
FileHandlingUtils.m	
General.c	
Index.c	
OSPanel.m	
Prefs.c	
StringUtils.m	
SystemDirectoryCopyURL.m	
32-bit Runtime	Comments
AppThings.c	mostly supplanted by CocoaUI
ConsoleApp.c	conditionally included from FBtoCConsole.incl. No longer supported
EditFields.c	essentially replaced by CocoaUI - remains available for 32-bit builds
FilesDollarFunction.c	supplanted by OSPanel. Obsolete/removed in current FB builds

 20170919 FB 5.7.105

[ttp://www.4toc.com/fb/index.htm](http://www.4toc.com/fb/index.htm) [you will need to reload the page if your browser cached the page mentioning FB 5.7.104]

FB 5.7.105 requires OS X 10.6 or higher

Introduction to New Containers

This release's container update could prevent your app from compiling and could cause it to malfunction.

Continuing the on-going work to move the FB runtime to 64-bit, FB containers were updated to remove Handles and use pointers.

Because containers are now based on pointers, container usage relying directly on Handles needs updating to work correctly.

Creating a container directly from a Handle won't work but dereferencing the Handle and using a helper function (see below) will.

Creating/modifying containers is basically the same except for the following:

- a. Toolbox Handle-involved calls (Set/GetHandleSize, PtrToHand etc.) cannot be used to manipulate containers.
- b. Reading/Writing to/from a container changes slightly because a container is a pointer
- c. Passing container pointers to other functions changes slightly.
- d. Code **CANNOT** dereference containers directly. The helper functions **MUST** be used (see 'To/From Pointer/Container' below)

Container size is limited based on whether the app is 32-bit(approx. 2 GB) or 64-bit(> 2GB).

- a. 32-bit user interface widgets like edit fields might have size limitations
- b. using 32-bit edit fields (or any other 32-bit UI widget) precludes a 64-bit app.

Containers and Edit fields. Unchanged

- | | |
|---|---|
| 1. Put container contents in an edit field. | <code>edit\$(_myEditField) = #gC</code> |
| 2. Load container from edit field. | <code>gC = edit\$(edit_field_id)</code> |
| 3. Load edit field from container. | <code>edit\$(id) = #gC</code> |

To/From Pointer/Container. New

- 1.
2. // get a pointer from a container.
3. Replaces handle syntax: `'myHandle = [@gC]'`
- 4.
5. `fn ContainerFromPointer( @gC, p, size )` // load container from pointer.
6. Replaces handle syntax: `'gC = &h'`

Both may be found in Util_Containers.incl

Old Handle syntax NOT supported

1. Loading containers from handle data. Use **fn ContainerFromPointer** instead of `gC = &h`
2. Loading a handle from a container. Use **fn ContainerToPointer** instead of `h = [@ gC]`
3. Adding TEXT resource data to a container doesn't work. There is no replacement. `gC = %128`
4. Handle toolbox calls won't work on containers (i.e. DisposeHandle, NewHandle, GetHandleSize, SetHandleSize).

File I/O to/from a container

1. NEW - reading using FB file I/O


```

    dim as pointer p
    open "I", 1, @url
    size = lof( 1, 1 )
    p = fn malloc( size )
    read file #1, p, size
    fn ContainerFromPointer( @gC, p, size )
    fn free( p )

```

2. NEW - writing using FB file I/O

// Create a pointer from a container and write the file from the pointer.

```

    dim as pointer p
    dim as container gC
    open "O", 1, @url
    size = len( gC )
    p = fn ContainerToPointer( @gC )
    write file #1, p, size
    fn free( p )

```

3. New I/O functions from Util_Containers.incl don't require FB I/O statements

```

    fn ContainerWriteToURL( @gC1, url ) // Write
    fn ContainerReadFromURL( @gC2, url ) // Read

```

Note: "Util_Containers.incl" must be in the project if these new functions are used.

Most of the FB container statemments and Util_Container.incl have the same usage even though internals have changed

Statements and functions with same usage are:

right\$\$, left\$\$, mid\$\$, len, asc, ucase\$\$, string\$\$, FBCompareContainers(),

Util_Containers.incl continues to have some functional overlap with FB keywords.
Those functions are listed below with FB equivalents

fn ContainerUppercase	ucase\$\$
fn ContainerInsertPascalString	mid\$\$ statement
fn ContainerFindPascalString	instr
fn ContainerAppendPascalString	gC += "some pascal string"
fn ContainerFindAndReplacePascalString	instr followed by mid\$\$ statement

CocoaUI Updates and Additions

Application

New:

```

AppKeyWindow
AppMainWindow
AppIsHidden
AppHide
AppUnhide
AppUnhideWithoutActivation
AppIsActive
AppActivateIgnoringOtherApps
AppHideOtherApplications
AppUnhideAllApplications
AppShowHelp

```

Box

New:

```

BoxSetTransparent
BoxSetTitleFont
BoxSizeToFit

```

Button

New:

```

ButtonSetTitleColor
ButtonSetAttributedTitle
ButtonSetAlternateTitle
ButtonSetPeriodicDelay
ButtonSetAttributedAlternateTitle

```


ButtonSetAlternateImage
ButtonSetTransparent
ButtonSetAllowsMixedState
ButtonSetNextState
ButtonHighlight

ColorWell

New:
ColorWellActivate
ColorWellDeactivate
ColorWellDrawWellInside

Control

New:
ControlSizeToFit
ControlSetFont
ControlSetFontWithName
ControlTakeStringValueFrom

DatePicker

New:
DatePickerSetBackgroundColor(
DatePickerSetTextColor(
DatePickerSetLocale
DatePickerTimeInterval
DatePickerSetTimeInterval
DatePickerMinDate
DatePickerMaxDate

Dialog

Changes:
1. Dialog events get/set functions consolidated, e.g. CocoaWindowEventXxxx, TabViewEventXxxx, SplitViewEventXXX, etc. are all now DialogEventXxxx.

2. on dialog fn now receives window content view _viewDrawRect event for CG drawing. Obtain the CGContextRef with DialogEventCGContext and content view rect with DialogEventRect.

CocoaUI Header files

Changes:
Removed 'Cocoa' prefix from filenames

ImageView

New:
ImageViewSetCGImage

LevelIndicator

New:
LevelIndicatorMinValue
LevelIndicatorMaxValue
LevelIndicatorWarningValue
LevelIndicatorCriticalValue
LevelIndicatorSetStyle

Changes:
1. Style param removed from LevelIndicator statement. Style is now set by LevelIndicatorSetStyle function.

Menu

Changes:
MenuItemEventSetBool renamed MenuEventSetBool

NibView

Changes:
viewIdentifier param now optional. If this param is omitted, the first NSView found in the nib will be used.

NibWindow

Changes:

wndIdentifier param now optional. If this param is omitted, the first NSWindow or NSPanel found in the nib will be used.

SecureTextField

New:

SecureTextFieldSetEchosBullets

SplitView

New:

SplitViewAdjustSubviews

Text

New:

TextString

TextField

New:

TextFieldSetTextColor

TextFieldSetBackgroundColor

TextFieldSetFormat

TextView

New:

TextViewSetBackgroundColor

TextViewSetDrawsBackground

TextViewSetUsesRuler

TextViewSetRulerVisible

TextViewSetUsesInspectorBar

TextViewSetSelectedRange

TextViewSetContinuousSpellCheckingEnabled

TextViewSetGrammarCheckingEnabled

TextViewStartSpeaking

TextViewStopSpeaking

TextViewSetUsesFontPanel

TextViewSetUsesFindPanel

TextViewSetUsesFindBar

TextViewTextStorage

TextViewSetRichText

View

New:

ViewWithTag - returns view ref for issuing direct obj-c calls

SInt32 ViewSuperview

Sint32 ViewWindow

void ViewSetFrameOrigin

void ViewSetFrameSize

void ViewSetFrameRotation

CGRect ViewBounds

void ViewSetBounds

void ViewSetBoundsOrigin

void ViewSetBoundsSize

void ViewSetBoundsRotation

CGFloat ViewAlphaValue

void ViewSetAlphaValue

CGFloat ViewFrameCenterRotation

void ViewSetFrameCenterRotation

void ViewPrint

void ViewSetNeedsDisplay

void ViewRotateByAngle

Window

New:

WindowWithTag - returns NSWindow ref which can be used for issuing direct obj-c calls

WindowSetAspectRatio

WindowSetContentAspectRatio

WindowSetDocumentEdited

WindowDocumentEdited
WindowToggleFullScreen
WindowSetAllowsToolTipsWhenApplicationIsInactive
WindowSetTitlebarAppearsTransparent
WindowSetTitleWithRepresentedFilename
WindowSetTitleWithRepresentedURL
WindowAttachedSheet

Changes:
Previously, only non-Cocoa apps would subclass window content views in order to draw focus rings around fields. Now, Cocoa apps also subclass the content view and intercept the drawRect method for CG drawing. Obviously, focus ring drawing is not applied to fields in Cocoa apps.

NibWindow
Changes:
A negative tag value in the nibwindow statement now works same as cocoa window statement (loads window invisibly).

General
Changes:
Functions requiring a higher macOS than set in Base SDK no longer silently fail and now report "implicit declaration of function 'xxxx'..."

Fixes & Additions
[01] Added InstallCocoaUIScripts app to FB Help menu
[02] Writing data in "R" mode could truncate file length (FH_WriteData truncateFileAtOffset:)
- reported by B.G.
[03] FBDelay() within General.c had a value conversion issue
[04] FBLoF() within FileHandling.c. Added conditional compile for 32/64-bit to silence compiler warnings
[05] PrintContainerWnd() in AppThings.c updated to use pointers instead of handles.
[06] Util_String.incl updated to silence warning for containsString: method.
[07] MidStmt_Ctnr() in containers.c was not doing the stack pop. reported by Mark.
[08] New version of Quartz (1.14) and demos for use with CocoaUI or Carbon.
[09] Editor source updated to use new pointer-based containers
[10] "Unknown type name 'FBMenuEventInfo' when compiling -DDECARBONATE -m64 and CocoaUI.

20170625 FB 5.7.104

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.102]

FB 5.7.104 requires OS X 10.6 or higher

Summary
Note: This release contains one or two updates which may prevent your app from compiling.

Updates:

- Bug fixes.
- 64-bit updates.
- New project template updated to new window statement syntax.
- Run statement now accepts CFString path or CFURL. The old 'path\$' syntax is deprecated.
- Kill Field statement removed. Those needing to continue with this functionality should either use an older release or use the direct toolbox call: if (h) { if ((HGetState(h) & 0x20) == 0) DisposeHandle(h); }
- System() function updated. Many old selectors removed.
- New FB header, Util_RunningApplication.incl, provides information about running application(s). See FB Examples/CoreFoundation & Cocoa/Cocoa/Util_RunningApplication.incl example(s)/RunningAppDemo.bas.
- TWM updated to version 1.9.14.
- New FB Header, Util_UserDefaults.incl, is a functional replacement for Util_CFPrefs.incl.
- On a dialog _btnClick event, dialog(-1) now returns the button's window number.

More Details

[01] Run statement updated to Run a CFString path or a CFURLRef. Old syntax deprecated. FBHelp updated too. Brian/Bernie

[02] Delay statement updated to 64-bit. Uses nanosleep(). Brian

[03] Kill Field removed for same reasons Read/Write Field were in FB 5.7.99.

Those needing to continue with this functionality should either use an older release or the direct toolbox call:

```
if ( h ) { if ( ( HGetState( h ) & 0x20 ) == 0 ) DisposeHandle( h ); }
```

[04] System() function updated. Many old selectors removed. _aplActive updated to modern 64-bit code.

_scrnWidth, _scrnHeight, _sysVers, _crntDepth are the other remaining selectors. FB Help updated accordingly & FBtoC reports errors

for any other selectors.

[05] FBGetProcessInfo() is now conditionally compiled only in 32-bit compiles. This assures the old code won't be there for 64-bit compiles.

FBHelp updated to reflect #5 and #6

[06] New FB Header, Util_RunningApplication.incl, provides information about running applications (duh!).

See FB Examples/CoreFoundation & Cocoa/Cocoa/Util_RunningApplications.incl example(s)/RunningAppsDemo.bas

[07] FBKillPicture() for now is within #if !__LP64__ to make sure it is NOT included on 64-bit builds.

At some point all of the PICTURE commands need review. Either update to use a non-Quickdraw graphic or remove.

[08] Bug fixes for AddKeywordToSymbolTable() and AddSubKeywordToSymbolTable() Symbol Table Implementation. Brian

[09] TWM (The Window Maker) updated to version 1.9.14. Bernie

[10] A new Util_UserDefaults.incl functionally replaces Util_CFPrefs.incl and is 64-bit. Note: the older Util_CFPrefs.incl has older Handle code in it. Bernie.

[11] dialog(-1) now returns the window number on _btnClick events. Bernie

[12] Various code cleanup in General.c. AppThing.c Brian

[13] Some parenthesized coordinates in Window statement fail. Bernie

[14] Window attributes fixes for new Window statement when attributes are missing or NULL. Bernie

[15] New Project template updated to new Window statement syntax. Brian

[16] Save panel 'Cancel' button not honored when saving a new editor window. Bernie

[17] New reserved words introduced. Note that reserved words cannot be used as variable names and will fail when building with "Bad factor in expression in line xxxx". The most likely new words to cause a failure are: 'imageview', 'panel', 'scrollview', 'slider', 'toolbar' and 'view'. For more information on these new reserved words, see CocoaUI.pdf in FB's Help menu

Note: Containers are unchanged in this release but an upgrade effort to 64-bit is in progress. If raw pointer speed isn't required, a 13-July-2017 list post, "Feedback request FB's containers", suggests alternatives.

20170625 FB 5.7.102

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.101]

FB 5.7.102 requires OS X 10.6 or higher

New

[01] Window & Appearance Window statements changes: Brian/Bernie

(a) **Appearance Window** statement is **deprecated** and the translator warns and suggests moving to the Window statement.

(1) Otherwise same as FB 5.7.101.

(b) **Window** statement's parameters are identical to Appearance Window except the rectangle and window attribute have changed.

(1) Rectangle parameter: If used, must be a 'CGRect'; a QuickDraw 'Rect' is no longer accepted. (x1,y1)-(x2-y2) format still available.

(2) Window attributes: Are specified in a standard array instead of OR'ing into a integer. Also, the attribute constant names have changed.

[02] New Grand Central Dispatch (GCD) demo (at: FB_5_7_102_Examples/CoreFoundation & Cocoa/GrandCentralDispatch(GCD)/GCD Timer Demo)- Steve VV

[03] OSPanelSave now defaults to hiding the file extension - Bernie

[04] FB header, Tlxb CFString.incl, updated to include CFStringInitInlineBuffer and CFStringGetCharacterFromInlineBuffer which are designed for fast access to string characters. - Brian

[05] New CoreText include with demo now in FB_5_7_102_Examples/Text/CoreText/CTTextFrame - Bernie

[06] FB's "Printing Without a Window" feature now uses anti-aliased text for PRINTing. Implemented via QDSwapTextFlags() and srcOr text transfer mode.

This was added as a convenience for those who need to run a quick test. It does not upgrade general window code. FBers are still responsible for windows they create.

[07] FinderInfo example using CFURL added. See: FB_5_7_102_Examples/Files/Droplet apps/ FinderInfo replacement(CFURL-based) - Bernie

[08] An example showing how to configure an info.plist for app and document icons. See: FB_5_7_102_Examples/Miscellaneous/info_plist for doc & app icons - Bernie

Was discussed on the list on 8-June-2017 and 10-April-2017

[09] New CFString and CFNumber utilities contained in two new headers (Util_String.incl and Util_Number.incl)

Provides the power of Cocoa/Objective-C methods but wrapped in easy-to-use FB functions.

Examples are in: FB_5_7_101_Examples/CoreFoundation & Cocoa/CoreFoundation/Strings and Numbers

Note: the content of the older Util_String(s).incl was combined with the new Util_String.incl

Fixes/Updates

[10] Editor source files with forward '/' slashes in the names silently fail to save file contents. Brian

Within the Editor's source code, OSPanel was substituted for files\$ and any file name slashes converted to ':'(as NSSavePanel does) as noted by Bernie and Ken

The issue arises due to the the Editor code's need to concatenate a parent and file name URL.

Once/If FSSpec usage is replaced with CFURLRefs in the editor's global array, URL concatenation and the scan to check for forward slash (/) will be unnecessary.

[11] Tlxb MoreFilesX.incl (supports deprecated file calls and structures) has been removed from the Headers. All FBers should switch to more modern calls. Brian

[12] ConsoleWindow support removed. Recommend using NSLog.

[13] Launching the Editor by double-clicking a single source file sometimes fails if old TYPE/ CREATOR (specifically TYPE) is zero

Editor was incorrectly interpreting such a file as a package project file, and then ReadProjectIntoDataBase() fails because it isn't a project. Brian

[14] gNewWndPositionMethod global within Rntm Appearance.glbl was causing an "unknown type" error for '-m64 -DDECARBONATE' builds. Bernie.

Solution: Restore gNewWndPositionMethod's definition to within the '#if ndef __LP64' so it is only used in 32-bit builds.

[15] Open "0" now defaults to OSTypes of zero for Type/Creator. Also see Def Open below.

[16] Def Open now defaults to NULL OSTypes for Type and Creator and can be requested explicitly(e.g. Def Open "")

Def Open's FBHelp updated to reflect changes.

Note: Def Open is deprecated. Recommend not using it and instead specifying UTIs and file extensions for icons in the application's info.plist. Bernie/Brian

Apple Documentation includes UTIs that have been standard since OS X 10.4:

https://developer.apple.com/library/content/documentation/Miscellaneous/Reference/UTIRef/Articles/System-DeclaredUniformTypeIdentifiers.html#//apple_ref/doc/uid/TP40009259-SW1)

also: <http://tinyurl.com/jd7uj2l>

[17] Updated FBHelp and FBtoC to reject Read/Write Field (per change in FB 5.7.99) a little more gracefully. Brian

[18] FixSDK updated to version 1.2.5. Bernie

(a) Fixes a bug that caused some tableview buttons to vanish.

(b) Now can Export/Import SDKs to to/from an external directory(s). See FixSDK Help for more details.

(c) FixSDK now launches fully when the 10.6 SDK is not in Xcode (the previous version told the user to quit and install the 10.6 SDK).

If a 10.6 SDK is not installed, all buttons are disabled except for Import which allows the user to locate a 10.6 backup. Once installed, the buttons become enabled.

(d) Button enabling logic bug fix

(e) Alert sheets warn users that existing SDKs will not be overwritten

- (f) 'Show SDKs in Finder' file menu item.
- [19] Search Apple Headers(SAH) app updated to version 1.0.5. The SDK popup menu showed '0.0' due to Apple's use of alises in their SDKs. Fixed. Bernie
- [20] TWM (The Window Maker) updated to version 1.9.13 supports the new Window Statement syntax. Bernie
- [21] NSLog.incl updates: settings window is a popover instead of drawer if available. Also implemented a bug fix- Bernie
- [22] Introduced as an option in release 5.7.42 (February 2016), the following statements now require CFString titles and no longer accept pascal strings
 - (a) Appearance Window
 - (b) Window
 - (c) Appearance Button
 - (d) Button
 - (e) apple menu (also accepts CFArray of titles)
 - (f) Edit Field
 - (g) Menu statement
 - (h) Shutdown statement

20170425 FB 5.7.101

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.100]

FB 5.7.101 requires OS X 10.6 or higher

New

- [01] Util_Workspace now a standard FB Header - Bernie
 - a) Access to many NSWorkspace options in simple FB functions
 - b) See the examples at FBExamples/Files/Util_Workspace examples
- [02] Util_Array.incl now a standard FB Header
 - a) Useful helper functions call to some of NSArray's methods. Bernie
 - b) One cool function is ArrayWithObjects() and could be used to pass UTIs to the new OSPanelOpen() as Bernie showed on list on 21-April-2017
 - c) See the examples at FBExamples/CoreFoundation & Cocoa
- [03] Util_Dictionary.incl now a standard FB Header
 - a) Useful helper functions call to some of NSDictionary's methods. Bernie
 - b) See the examples at FBExamples/CoreFoundation & Cocoa

Fixes/Updates

- [04] OSPanelOpen/OSPanelSave's FBHelp received some minor modifications to clarify the type of parameters it accepts. Brian (pushed out in later update to .100 too)
- [05] FB 5.7.100 issues files\$ deprecation warnings and suggests OSPanelOpen/OSPanelSave. Brian
- [06] Files\$'s FBHelp updated to issue deprecation warnings. Brian
- [07] Minor updates to FBtoCConsole.incl to use OSPanel instead of files\$. Bernie
- [08] Minor update to MDArray.incl in FBExamples. Bernie
- [09] FBtoC printed a double "Warning:". Bernie
- [10] Apple's new file alias inclusion in its SDKs generates a Base SDK popup with a nameless first item. Bernie.
- [11] Editor file save crashes when user selects Cancel. Bernie
- [12] FBtoC crashes trying to execute showAlert() and its subordinate [NSAlert runModal]. Needs to run on main thread. Brian
- [13] Editor updated to use OSPanel for File==>Open... menu option. Modernizes code and allows it to remember the last directory the FBer opened when used next time. Brian
- [14] Minor updates to OSPanel's FB Help to clarify allowFileTypes are file extensions and UTIs and not OSTypes
- [15] VAList in FBExamples updated to use the C macros.

20170413 FB 5.7.100

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.99]

FB 5.7.100 requires OS X 10.6 or higher

New

[01] OSPanelOpen/OSPanelSave are new built-in FB keywords. They offer more options than files\$ but don't use the Carbon framework. Bernie with some nudges from Brian
[02] FBHelp updated to include OSPanelOpen/OSPanelSave. Bernie/Brian

Fixes

[01] 'redefined variable' errors if both Util_FileDirectory.incl & Util_FileManager used in same project/file. Bernie
[02] DEFSTR long incorrectly generates '8' instead of '4', so any functions using the gPSDefStrSize global receive the wrong size. Brian
[02] TWM - The Window Maker - builds graphical window and generates FB source - now included in FB5. Bernie

ALSO: the following copy of the on-list announcement contains good information:

Hello Everyone,

This release introduces OSPanel and its new FB keywords, OSOpenPanel and OSSavePanel. Since they are keywords, there are no extra includes to remember and are part of the standard FB runtime. They provide the functional equivalent of the old files\$ plus additional features. Here are some (but not all) new features:

- Designate a starting directory for the panel(dialog)
- Remembering directory from last open
- Designate tags
- Extending a dialog with your own controls
- Sheet use¹

Additionally, the new keywords are very flexible (parentheses and all parameters optional). Plus there are other subordinate new keywords that can be used to prepare the panel's appearance and other related needs.

The best way to learn about all the new keywords is exercise the demos at 'FBExamples/Files/OSPanel examples' and review the FBHelp.

Recommendation:

OSPanel is now recommended instead of files\$ or the (carbon-based) NavDialog calls and both of the latter are deprecated and should not be used going forward. files\$ is available in 5.7.100

Reason for change:

Even though files\$ was limited to using CFURLRefs in the last release, it continued to use Carbon-based NavDialog calls for the dialog UI. The change to OSPanel makes all the code Carbon-free.

1. Sheets attached to FB-created (i.e. appearance window) parent windows may exhibit detachment of the parent window in some specific scenarios and other undesirable window drawing artifacts. Apparently, not every OS releases exhibits this problem but the issue is rooted in a parent Carbon window (i.e. parent Cocoa windows are fine). The demos show two ways of doing sheets both with/without Cocoa windows.

20170320 FB 5.7.99

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.97]

FB 5.7.99 requires OS X 10.6 or higher

[01] FileHandling.c updated to 64-bit - Brian
All FB's I/O verbs no longer accept FSSpecs/FSRefs and only accept CFURLRefs. See Jan/Feb 2017 FB list discussions for details.

- a) the file-based version of the OPEN statement must specifically use the CFURLRef version (OPEN UNIX and OPEN "C" excepted because not file-based)
- b) READ/WRITE FIELD are removed and no longer supported due to use of deprecated Handles. Replacement options were discussed in Jan/Feb 2017 FB list discussions.
- c) FB Help's 'Appendix A - File Object Specifiers' updated to reflect removal of FSRef/FSSpec support and sample code updated.
- d) FB's Rename verb now accepts only CFURLRefs. Syntax is: rename urlForCurrentFile urlForNewFile.
- e) FB's Kill now accepts only a CFURLRef for the file name.
- f) FB's files\$ now only uses/returns a CFURLRef for each of the three major options listed in FB Help. FB Help updated to reflect changes.
 - 1) "ConsoleWindow"'s usage of files\$ updated to use URL. I don't like it because ConsoleWindow depends on an FSSpec for its TXNSave() use.
 - 2) New constants for files\$ _URLOpen, _URLFolder and _URLSave are preferred over their predecessors _CFURLRefOpen, _CFURLRefFolder and _CFURLRefSave but have the same values.
 - 3) OPEN/FILES\$ changes to Editor source to allow it to build with FB 5.7.99.
 - 4) FBtoC flags non-CFURLRef files\$ usage for the variable but does not check the mode constant; but runtime honors only valid modes.
 - a) Make sure to check files\$'s returned fileName for non-zero length. Zero indicates failure and possible bad mode constant.
 - g) Apps crashing with a "Bad file descriptor" message indicates a file i/o attempt which isn't allowed by the OPEN mode (i.e. opening in mode "I" and trying to WRITE the file).
 - h) A second attempt (either within the app or another app)to OPEN "N" on a file currently/already open in "N" mode has specific new behavior compared to prior FB5 versions:
 - 1) The second attempt process/code is given read-only access (essentially "I" mode) to the file.
 - 2) The FB runtime sends a "file already open with with write permission" (_opWrErr) error code which can used by the caller (must be trapped with 'on error' and "error")
 - 3) Even though opened for "I" after a request for OPEN "N", it must be closed or any subsequential exclusive OPEN requests will fail.
 - 4) FBers should always check the error status returned by any file OPEN and take appropriate steps.
 - 5) N.B. OPEN code only provides access; it does not provide any data integrity protection when there are concurrent readers and writers of the same file.

[02] Util_FileManager additions - Bernie

- a) File access functions written with only modern 64-bit code. Eventual replacement for Util_FileDirectory.
- b) Only supports CFURLRefs and does not support FSRefs or FSSpecs.
- c) Several demos available in the FB Examples
- d) Util_FileManager.incl includes Util_PathUtilities.incl and Util_URL.incl
- d) Util_FileDirectory is still available for those who need it but folks should plan to migrate to Util_FileManager.
- e) For fn FM_TrashItemAtURL (but not the other functions in the include). The FBer has two choices:
 - 1. Its current implementation uses a method requiring OS X 10.8+
 - 2. Another method, which is currently commented out and doesn't have the 10.8+ requirement, is possible but it is deprecated in OS X 10.11.
 Once you've decided which constraints are best for you, simply comment out the one you don't want and uncomment the other (or vice-versa).
 Consider using the user includes folder if maintaining it yourself.

[03] New/Changed Headers and Examples - Bernie, Steve VV, Brian

- a) A slick multi-dimensional Core Foundation array implementation handles all the details. See: FB_Examples/CoreFoundation and Cocoa/MDArray(multi-dimensional)
- b] Tl bx CFUUID.incl and Tl bx CFStringTokenizer.incl added to Headers
- c] CFMeasurement headers and demo added to FB Examples
- d] Superscript/Subscript demos added to FB Examples
- e) Util_Files.incl uses outdated (parameter block) file functions. N.B. IF YOUR CODE USES FUNCTIONS FROM THIS INCLUDE, IT WON'T BUILD.

[04] Default error function - Brian

- (i.e. when it isn't supplied via 'on error fn yourErrorHandler' in your own code) updated to include:
- a) error string and comment
 - b) N.B. Carbon calls GetMacOSStatusErrorString() and GetMacOSStatusCommentString() were NOT used. Cocoa's NSError used [NSError errorWithDomain:NSOSStatusErrorDomain

[05] NSLog.incl updates - Bernie

- a) silence a clang 'dealloc' warning (accidentally omitted in 5.7.97)
- b) Copy & Copy All items added to text view's contextual menu
- c) NSLogBeginEditing & NSLogEndEditing. Bernie's explanation follows:

Multiple NSLog calls in big loops can make it appear that your app is hanging. After issuing NSLogBeginEditing, the NSLog text view is only updated at the point NSLogEndEditing is called.

```
// - Example1 ---
```

On my machine, this takes about 95 seconds for the text to appear.

```
for i = 0 to 150000
  NSLog(@"string %d",i)
next i
```

```
// - Example 2 ---
```

Enclosing the code in an NSLogBeginEditing/NSLogEndEditing pair takes less than 4 seconds.

```
NSLogBeginEditing
```

```
for i = 0 to 150000
  NSLog(@"string %d",i)
next i
```

```
NSLogEndEditing
```

Fixes - Brian

[06] FBtoC Help, inadvertently omitted in 5.7.97, is back again.

[06] Editor scrolls source view to an incorrect line number but FBtoC's line number is accurate (captured value of global in local before doing a dispatch async)

[08] Comparing an FB container to a string constant resulted in string stack failure (changed gFBStk to SInt16 in Runtime.h and General.c so it can go to -1 instead of 65535)

[09] FBtoC's Settings dialog would not show up when requested via the Editor's preference pane option. Thanks to RC and BW for noting and reminding.

Internal: Added source for "Error Codes" app to FB project source

ALSO: the following copy of the on-list announcement contains good information:

Hello Everyone,

Please read the following introduction before downloading and replacing your current FB; it is important to know the impact. Download link at the bottom.

Introduction

Unlike some FB5 releases, this one contains several changes which impact your current FB source code. Depending on your code, it may not compile with this FB version.

Internally, all of FB runtime disk-based file I/O system (with some specific exceptions which will be noted later) was updated and is now 64-bit compatible. In some cases the updates impact the operation and syntax of specific FB file I/O keywords. Most FB file I/O keyword's operation and syntax are exactly the same while a few have slightly different syntax/requirements and behavior. It is important for FBers to understand these impacts, assess the work involved, and decide how/when to implement this new FB release. Like usual, old releases and the prior release ([5.7.97](#)) are available. Hopefully, this moves us closer to a "Future" BASIC. The design goal was to keep the language in familiar territory with the same keywords while upgrading the supporting runtime.

Overview of Changes to FB File I/O

As usual, the [Release History Notes](#) detail the changes but here is an overview.

[A] File-based versions of the OPEN statement(open modes "I", "O", "A", "R" and "N") must specifically use the CFURLRef version. OPEN no longer supports FSSpec or FSRef.

[B] OPEN "N" now provides an automatic read-only open when a file is already open in "N" mode. It notifies the code of the automatic selection of "I" mode, when "N" was requested, via FB's standard ERROR functions. See code below at note:¹

[C] READ/WRITE FIELD are removed due to use of deprecated Handles and potential links/associations with old code.

Replacement options were discussed in Jan/Feb 2017 FB list discussions. See this link for a possible replacement strategy: <http://freegroups.net/groups/futurebasic/0::67289read.html>

[D] FB's FILES\$ now uses/returns only a CFURLRef for each of the three major options(i.e. selecting a file, a folder, or save location/name)

[E] Util_Files.incl uses outdated (parameter block) file functions. N.B. IF YOUR CODE USES FUNCTIONS FROM THIS INCLUDE, IT WON'T BUILD.

In all cases this include's functions can either be replaced with similar URL functionality or the functionality is obsolete (Resource fork stuff). If you can't find modern replacements, please ask on the list. One replacement example: fn FSSendFileToTrash(sourceObj as ^FSSpec) can be replaced with one of the new Util_FileManager.incl code. (see FB Examples==>Files==>Util_FileManager Demos==>TrashItemAtURL

[F] Coding errors can cause crashes because the error isn't found until it runs.

For example:

(1) A "Bad file descriptor" crash message indicates a file i/o attempt which isn't allowed by the OPEN mode (i.e. opening in mode "I" and trying to WRITE the file)

(2) Using the wrong fileId. i.e. opening file #1 for read and then trying to read file #2 when fileId #2 doesn't exist.

The [Release History Notes](#) detail many other additions and changes and FBers are strongly encouraged to read and understand their impact.

What didn't change in FB File I/O

[F] Even though their underlying code changed a lot, most FB file I/O verbs/keywords use the same syntax. For example,

Error, Close, Write#, Input#, Print#, Record, LOF, EOF, Rec, Loc, Read File, Write File, Append, Open "C" and Open "UNIX" all have the same syntax. Such is the benefit of an abstracted high-level language.

[G] The relative performance and usage recommendations remain the same. For example, using read# with multiple variables generates more code(potentially a lot more) than doing a read file# into a FB record (C structure) or pointer. Read# generates one physical read per variable versus one read for the entire FB Record in Read File#. Here is an example with an FB record reading four variables with Read File vs. Read:

```
begin Record myRec
dim as short    a
dim as long     b
dim as Str255   c
dim as double   d
end Record
dim as myRec      aRec
```

This FB: read file #2, @aRec, fileSize
Generates this C: FBReadFile(2, (void*)(void*)&aRec, false, fileSize);

This FB: read #2, aRec.a,aRec.b,aRec.c;13,aRec.d
Generates this C: aRec.a = FBReadSwapShort(2);
 aRec.b = FBReadSwapLong(2);
 FBReadString(2, (char*)&aRec.c, 13);
 aRec.d = FBReadSwapDouble(2);

[H] This FB release brings lots of new headers, updates and bug fixes. Most timely is the new addition of Util_FileManger to assist with the new OPEN and FILES\$. Thank you Bernie.

Notes:

1. Code below shows interception of an OPEN "N" notification and how to determine if "I" mode was assigned because file is in use.

```
//-----  
include "NSlog.incl"  
  
dim as CFURLRef      url  
dim as short        ioError : ioError = 0  
dim as CFIndex       fileSize  
dim as CFStringRef  s  
dim as dim as        p  
  
s = @"/Users/brians/Desktop/mytextfile2.txt"  
url = fn CFURLCreateWithFilePath( NULL, s, _kCFURLPOSIXPathStyle, _false )  
  
on error end  
error = _noErr  
open "N",1, @url  
ioError = error  
error = _noErr  
// _opWrErr ( -49 ) file already open with with write permission*/  
// in this case another process has the same file open in mode "N"( since this process does not  
) ,  
// so the runtime opens the file for you in read-only mode ( like mode "I" ) and notifies  
// the FBer of this action by sending back the _opWrErr code. Brian 20170303  
  
select switch ( ioError )  
  
case _noErr  
    // No problem, your code has the file open in "N"  
  
case _opWrErr  
    // other code ( or code in another process ) already has the file open in "N"  
    // so your code now has this file open in "I" mode  
    NSLog(@"ioError = %d", ioError )  
    fileSize = lof(1,1)  
    p = fn malloc( fileSize )  
    read file #1, p, fileSize  
    s = fn CFStringCreateWithBytes( NULL, #p, fileSize, _kCFStringEncodingMacRoman, _false )  
    free( p )  
    NSLog (@"%@",s )  
    CFRelease( s )  
    Close #1  
end select  
  
CFRelease( url )  
  
RunApplicationEventLoop  
//-----
```

General note: Several FBers contribute to the update effort and the release notes now identify the persons mostly (approximately 95%+ of the effort) responsible for the coding effort in each area, even though others not mentioned might have contributed.

20161228 FB 5.7.97

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.94]

FB 5.7.97 requires OS X 10.6 or higher

[01] FBtoC is now a 64-bit app. This means we have a translator that will run even if Apple removes Carbon - Brian

Bonus: removal of more runtime code reduced FBtoC's disk size.

[a] Translate/Compile runs on a separate queue/thread(libdispatch driven) to reduce/eliminate FBtoC's previous 'beachball' cursor.

Note: Macs with fewer than two cores will not see all the benefits of concurrent multi-thread processing but the 'beachball' should be (mostly) gone.

[03] Remaining 'route_toBuffer' and associated Handle management code replaced with CFMutableStringRef(one for errors and one for code).

[04] NSTabView convenience functions added to Util_UI.incl header.

[05] FBtoC app is codesigned (FB editor is not codesigned) for Apple Gatekeeper compatibility.

[06] FBtoC now copies story board files created in Xcode into the app package. See Storyboard Demos in FB Examples/Cocoa.

[07] Minor update to NSLog.incl silences a clang 'dealloc' warning.

Fixes - Brian

[08] Long constants (> 255 characters) within BeginCxxxx/EndC would be flagged as errors (string too long).

[09] Multiple successive invalid ids passed to Edit\$ statement in user program overflows gFBStk's maximum value.

[10] fn AddConstantsFromResourcesToSymbolTable incorrectly converted old built-in constants to unsigned long instead of signed long. This resulted in "implicit conversion" compiler warnings and could cause bugs.

[11] Building an empty (or with one line like 'print') single source file over twenty times would crash. Forty successive tests succeeded without a crash. *Might* be fixed.

[12] fn WriteCodeToURL would crash CFReleaseing a null CFURLRef. The problem is the path passed to WriteCodeToURL() was longer than the 1024 characters allowed, so the conversion to CFURLRef fails. Same potential issue fixed in fn WriteCharToURL too. N.B. note the path length limitation.

Coding and implementation: Bernie, Brian and Deep

2016xxxx FB 5.7.95

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.94]

FB 5.7.95 requires OS X 10.6 or higher

[01] FBtoC now copies story board files created in Xcode into the app package. See Storyboard Demos in FB Examples/Cocoa.

[02] Minor update to NSLog.incl silences a clang 'dealloc' warning. In Headers.

[03] Bug: fn WriteCodeToURL would crash CFReleaseing a null CFURLRef.

The problem is the path passed to WriteCodeToURL() was longer than the 1024 characters allowed, so the conversion to CFURLRef fails. Fixed

Same potential issue fixed in fn WriteCharToURL too.

N.B. 5.7.94 and 5.7.95(internal) are the last FBtoC versions to be all FB source code. Most of 5.7.97 is FB source but it was converted to Xcode

before 5.7.97 was released. Bottom line: if you want source for 5.7.97+, look at the Xcode source and NOT the FB source. Brian 20170111

Coding and implementation: Bernie, Brian and Deep

20160916 FB 5.7.94

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.69]

FB 5.7.94 requires OS X 10.6 or higher

[01] Updated fn GetNumber in Lexical Analysis - internally it's all CF but it still returns a pascal string. A start on the last conversion phase, lexical analysis.
[02] Remaining left\$, right\$, mid\$, ucase\$ and instr usage converted to CF. One 'hex\$' remains.
[03] Consolidated five functions (fn GetLeftParenthesis, right parens, brackets, equals, comma) into one function
[04] Updated GetLexeme_Priv in lexical analysis to CF and remove all the 'exit "GetLexeme_Priv"' so it will generate a normal C switch statement
[05] CFStringFromPStr() changed to accept pointer to Str255 instead of just Str255. Saves an extra copy for every call to it.
[06] Code buffers updated directly instead of using intermediate string. Translation speed improves 40 to 50% in some cases.
[07] New FixSDK app updated to handle Xcode 8 SDK changes. Thanks to Bernie for the research and updates.

Fixes

(08) An 'xref @' variable translates incorrectly. Fixed. (thank you to Paolo C. for reporting)

Coding and implementation: Bernie, Brian and Deep

20160815 FB 5.7.69

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.49]

FB 5.7.69 requires OS X 10.6 or higher

Fixes

(01) After reporting user code errors, FBtoC would lock up and not finish. Fixed. (thank you to Peter B. for reporting)
(02) Crash when launching FBtoC from desktop(i.e. without the Editor) with no build_goodies/ Headers files. Fixed
(03) HexString() incorrectly truncated when honoring some 'defstr' (i.e. byte, word, long) settings. Fixed.

FBtoC internal code updated to 64-bit

(04) fn EmitData updated to use CF and remove FB's dynamic array
(05) fn GotoStatement updated and its subordinate fn FixLabel (now named fn FixLabelCreateCF)
(06) fn GosubStatement updated and its subordinate fn FixLabel (now named fn FixLabelCreateCF)
(07) fn ReadQuotedString updated to CF and renamed fn ReadCreateQuotedString
(08) Miscellaneous global pascal strings used in 'Exit' code changed to CF mutable strings - see fn ClearMiscGlobalStrings
(09) Prefix labels for both generated and source (i.e. 'LL' and 'L') changed to CF - see kprefixForLabelFromFBSource/kprefixForGeneratedLabel
(10) All 'Exit' related code updated to CF including fn UniqueLabel
(11) Changes to fn BSComparePStr() and fn BinarySearch() to use CFStrings. Superficial changes at the moment.
(12) Many fn InString calls replaced with fn CFStringFind for slight performance boost.
(13) Some Unix functions, such as fn CopyUnixCommandFirstLineResponse, updated to not use FB runtime
(14) Replaced Handles with CFStrings for fn StringExpressionContinuation and other 'String Expression & Assignment' functions
(15) FSp LongName File Utilities removed and its only active function, fn FSpTrashObject, moved to Cocoa File Utils.incl
(16) SendTextFragmentToCommentBuffer() updated to Core Foundation
(17) Various 'Copy Resources Phase' FNs changed to CFStringGetCharacterAtIndex() for small speed gains.
(18) Prelexical State: PrelexicalState record changed to use CFStringRef instead of Str255. PushPrelexicalState(), PopPrelexicalState() changed to manage the CFStringRef memory appropriately.

Coding and implementation: Bernie, Brian and Deep

20160711 FB 5.7.49

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.48]

FB 5.7.49 requires OS X 10.6 or higher

- (1) Tlhx CFSet.incl, Tlhx CFBag.incl and Tlhx AppThings.incl added to Headers
- (2) CF replacement functions for right\$, left\$, mid\$, instr and hex\$ now easily accessible via new header file, Util_Strings.incl.
- (3) Crash converting old project file format to new. Fixed.
- (4) Error reporting fn was releasing the error string too early (fn BuildStringAndReportError) which could cause unexplained crashes. Fixed.
- (5) Build Settings: edit buttons for 'Path to Developer folder' and 'Compiler Options' were inoperative. Fixed.
- (6) Several translation bugs fixed related to 'def fn using', dynamic arrays and other issues. These were coding errors made while converting FBtoC source. Mark(LabHelper X) and Peter's(Drafting) source were instrumental in identifying issues.
- (7) Smart tabs: If the user enters an initial tab value on creation, it failed to select the tab correctly: Fixed.

```
appearance button _cTabs,, 2,,, @"Tab 1;Tab 2", @r, _kControlTabLargeNorthProc
```

Coding and implementation: Bernie, Brian and Deep

20160527 FB 5.7.48

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.47]

FB 5.7.48 requires OS X 10.6 or higher

Thanks to Bernie's prolific coding efforts, tab control and radio group implementation is totally easy. SmartTabs/SmartRadioGroups are implemented in the FB runtime, so all the messy work is done for you there.

If you'd like to try it, download the example at: FB_5_7_48_Examples/Controls/SmartTabs+RadioGroup1

Important: Please Note (N.B.)

FB 5.7.42, in February 2016, announced the next FB release will not support pascal strings in appearance window, appearance button, apple menu and menu statements. That change is still planned but there is no firm schedule yet. Those statements continue to support both pascal and CF strings in 5.7.48 but FBers should be making changes if not done already.

Coding and implementation: Bernie, Brian and Deep

20160527 FB 5.7.47 rev 2

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.47]

FB 5.7.47 requires OS X 10.6 or higher

- (1) read # and write # were not translating correctly causing the build to fail with compile errors. Fixed.
- (2) Two 'Quit' items on FBtoC's menu negatively impacted operation. Fixed
- (3) Added CFRelease(msg) to STACK_PUSH() to quiet the analyze warning
- (4) Build Setting, 'Check Array Bounds', failed to generate compilable C code in some scenarios. Fixed.

Coding and implementation: Bernie, Brian and Deep

20160516 FB 5.7.47

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.46]

FB 5.7.47 requires OS X 10.6 or higher

(1) New Stack_Push() function replaces older macro because clang was emitting hundreds of warnings about potential unsequenced errors. Function not only avoids any sequence errors but checks stack pointer to make sure it is within the stack array bounds. If not, it alerts the user and quits FBtoC.

(2) FBtoC would spew errors after multiple(20+) builds of same project(see 5.7.45 notes #3). Stack pointer not being decremented in FBOpenUnix(). Fixed.

Coding and implementation: Bernie, Brian and Deep

20160515 FB 5.7.46 - development only

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.45]

FB 5.7.46 requires OS X 10.6 or higher

(1) SmartTabs plus SmartRadioGroups - temporarily turned off until serious bug can be fixed.

Coding and implementation: Bernie, Brian and Deep

20160504 FB 5.7.45 - development only

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.44]

FB 5.7.45 requires OS X 10.6 or higher

(1) Tl bx ControlDefinitions.incl updated to fix errors in GetDataBrowserUserState() and SetDataBrowserUserState()

(2) Four code buffers(declarations, main, functions and comments) converted to CFArrays of CFMutableStrings (formerly arrays of allocated Handles).

(3) Crash building FBtoC source with "source combining" turned off resulted in crash when memmove() overwrote the end of the array and corrupted another variable.

Increasing gFBStrStk's size from 32 to 128 via kMaxStringStackLevel constant *bypasses* the bug. Bug was found/fixed in 5.7.47

(4) Edit Field's FB Help updated to reflect option to use Core Foundation string in title/text

Coding and implementation: Bernie, Brian and Deep

20160411 FB 5.7.44 - development only

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.43]

FB 5.7.44 requires OS X 10.6 or higher

(1) build_goodies now contains AppThings.m to support new FBFullStop()

Coding and implementation: Bernie, Brian and Deep

20160408 FB 5.7.43 - development only

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.42]

FB 5.7.43 requires OS X 10.6 or higher

- (1) Project Template (i.e. the default project when File>>>New Project is selected) now uses Core Foundation strings.
- (2) DEFINEDINCOCOA constant now recognized
- (3) Util_UI.incl header updated
- (4) Printing folder added to FB Examples. Contains "Print info using NSPrintInfo" and other Cocoa and Carbon (i.e. "PM") printing examples.
- (5) Bug fix: FBtoC crashed in fn SendHandleFragmentToOutputStream when an 'end if' was used to terminate an fn and a local variable wasn't dimmed.
- (6) Getter/Setter for gCurrFileName and other related code updated to use CFString
- (7) ChkBounds() updated to use CFString

Coding and implementation: Bernie, Brian and Deep

20160229 FB 5.7.42

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.41]

FB 5.7.42 requires OS X 10.6 or higher

Thanks to Bernie's prolific coding efforts, we now have new title syntax for some primary FB Statements.

- (1) Completely rewritten version of 'Search Apple Headers' application
- (2) Similar to the Menu statement changes of FB 5.7.39, the following FB statements now accept either Core Foundation(CF) or Pascal title strings:

- appearance button
 - appearance window
 - button
 - window
 - apple menu (also accepts CFArray of titles)
 - Edit Field (FB Help does not reflect this capability yet)

- (3) The following helper functions are available to get/set CFString text in the menu, appearance window and appearance button statements:

- For Menus: MenuSetTitle
 - fn MenuCopyTitle
 - MenuItemSetText
 - fn MenuItemCopyText

- For Windows: WindowSetTitle
 - fn WindowCopyTitle

- For Buttons: ButtonSetText
 - fn ButtonCopyText

- (4) FB Help now reflects #2's syntax

- (5) CFMenuWndBtnTitles.bas demonstrates #2's CFString capabilities and may be found in FB_5_7_42_Examples/Dialogs and Windows/

- (6) Deprecated usage (see N.B. next) based on #2 fixed in FB Editor code, the header FBtoCConsole.incl and (hopefully) the all the example code in FB_5_7_42_Examples

Important: Please Note (N.B.)

FB's next release will not support pascal strings in appearance window, appearance button, apple menu and menu statements, so FBers are strongly encouraged to upgrade those statements to the new CFString syntax.

Apple has not deprecated Pascal strings(but are discouraged due to limitations) but FB's Runtime Pascal support uses older 32-bit code. The goal is to nudge our code toward 64-bit compatibility as much as possible without disrupting FBers; small incremental runtime changes should make this easier on everyone.

As a reminder, FB 5.7.42 displays a deprecation message during a build/run: "Warning: Pascal string title is deprecated; use Core Foundation strings instead in line 97 of CFMenuWndBtnTitles.bas"

Coding and implementation: Bernie, Brian and Deep

20160214 FB 5.7.41

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.40]

FB 5.7.41 requires OS X 10.6 or higher

- Fixed: framework included from /Library/Frameworks not working
- Header, Util_UI.incl, added to include Cocoa nib loading & corresponding demo, CocoaNibDemo, added to FB Examples/Cocoa
- FB_5_7_41_Examples/CoreFoundation now includes a demo of replacement CoreFoundation/Foundation string functions(i.e. CF functions to replace, left\$, right\$, mid\$, instr, space\$, string\$)
- Help menu now includes links to the FBtoC web page and FutureBASIC mailing list on associate.com
- Better help message for first-time users seeing the "Could not get read/write access to preferences" message.

201600121 FB 5.7.40

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.39]

FB 5.7.40 requires OS X 10.6 or higher

- Fixed: FBtoC crashed when launched in OS X 10.7
- Added to Tlxb StdCLib.incl: toolbox fn memcpy(ptr dest, ptr src, UInt32 n) = ptr

20160108 FB 5.7.39

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.38]

FB 5.7.39 requires OS X 10.6 or higher

(1) Bernie implemented new CFString/CFArray support for the Menu statement. There is a nice demo in FB Examples/Menus. Thanks Bernie.
(now get back to work on that new editor! ;-)

(2) help folder message eliminated for now

20160105 FB 5.7.38

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.14]

FB 5.7.38 requires OS X 10.6 or higher

Introduction: The following notes don't adequately capture the significant depth and breadth of FB's (mostly all in FBtoC) changes but hopefully conveys a little of the effort involved.

-With this release, approximately 50%+ of FBtoC is 64-bit (compare to approximately 10% for 5.7.14)

-The editor returns with only minor updates to maintain FBtoC compatibility (Style files, .rsrc resources).

-Lexical code is the next area for FBtoC 64-bitness.

-Several FBers are working on a new 64-bit editor and results are encouraging. Please wish them luck & give thanks; it's a lot of work.

SUMMARY

Items of interest for FBers:

- FBHelp's Begin/end globals now describes the possibility to create C static local variables in local functions.

- Milestone: FBtoC now uses CFString file paths and urls exclusively; there are no FSSpecs.

- FB's generated apps now includes the NSHighResolutionCapable key set to true in their info.plist

- Mid\$ function updated to honor numChars variable. If numChars is zero, Mid\$ returns zero characters and not the whole string as it did before.

- RUN command updated to accept:

- (a) a CFString literal

- (b) a CFString variable

- (c) CFURLRef variable

- (d) FB Help for Run command updated to reflect new syntax shown

- FB Help Appendix A: File object specifiers updated to recommend CFURLRefs and not FSRefs or FSSpecs.

- 'kill resources' statement isn't supported and is ignored.

- FBHelp's Files\$ doc corrected for options (2) Selecting a Folder and (3) For Selecting a File Name and Folder where a file may be Saved.

- this was noted by Bernie and RC in a list discussion 18-June-2013.

- Replaced use of old '.rsrc' resources in editor because FBtoC doesn't process .rsrc files

- fn CompilerPath, fn CopyUnixCommandFirstLineResponse and some others changed to accept CFString input

- Support for .rsrc old style resource fork files removed.

- FBtoC used to combine data from all resource fork files and create an AppName.rsrc with only a data fork. This is now gone

- Removed copying of 'Localized.rsrc' as it is no longer needed.

- 'rsrc' resource processing fns removed

Major Areas of Code Conversion to 64-bit

- All file I/O related functions are now use Core Foundation or Foundation. This includes all cached file paths, constants and file reads.

- FB Dynamic arrays replaced with CFMutableArrays

- All "Output Stream" which is all code to copy resources, include files, build files, manage PCH and more

- All error handling

- All TranslateBuild Utilities, including all calls to unix utilities and building buffers for those utilities (i.e. AppendToBuf())

- All FSSpecs removed from all build-related code. Now all CFURLs and CFString file paths.

List of all (mostly) Enhancements:

- Mostly INTERNAL UPDATES ONLY(blue-colored text indicate items of general interest - reverse chronological order with most recent changes first)

- FBHelp's Begin/end globals now describes the possibility to create C static local variables in local functions.

- FB dynamic array sFBRuntimeFunctionsToStrip in Symbol Table Implementation replaced.

- Dynamic array, sIncludedFileNames in Prelexical State replaced

- More FBtoC internal code updates

- [01] FB dynamic arrays gCRuntimeIncludeFiles, gFrameworks, gFrameworkPlusHeaders and associated processing in "Conditional Inclusion" replaced with CFMutableArrays

[02] Replaced FB string constants in Conditional Inclusion file (and code impacted by them) with CF versions

[03] fn WriteTranslatedCodeToFile now accepts CFString input

[04] ReadSourceFile now accepts CFString input

[05] fn CopyProjectFileURL replaces fn GetProjectSpec

[06] Milestone: FBtoC now uses CFString file paths and urls exclusively; there are no FSSpecs.

[07] Remaining FB dynamic arrays in Write Translation file converted.

[08] FB's generated apps now includes the NSHighResolutionCapable key set to true in their info.plist

[09] fn RenameOutput and associated arrays converted

[10] Cleanup of fn SendMsgToEditor - seven lines of code reduced to one using Foundation instead of CF

[11] fn EmitData converted - this handles FB's DATA statements and associated READ and RESTORE support

[12] fn WriteTranslatedCodeToFile - more work

[13] fn SaveSourceUserHeaderSearchPaths replaces fn SaveFBSourceFolderReference and uses a CF mutable array instead of an array of FSSpecs

[14] fn SetUpFilePathsForThisProject now uses CFStrings.

[15] fn WriteTranslatedCodeToFile - updated to use CFStrings and MoveCFStringToRouteBuffer() instead of FB's PRINT

[16] ReadSourceFile processing converted but still accepts Str255 input

[17] WriteCTypeTableFile converted

[18] WriteHandleToFile.(now named: SplitHandleToHeader_C_Line) converted

[19] fn SaveSourceUserHeaderSearchPaths replaces fn SaveFBSourceFolderReference. It saves CFString search paths in a global array sCFSearchPaths

[20] Mid\$ function updated to honor numChars variable. If numChars is zero, Mid\$ returns zero characters and not the whole string as it did before.

[21] fn LastFileExists/fn BuildTempExists now accept a CFURLRef. Calls using it modified to send CFURLRef.

[22] fn ProcessOpenedFile now accepts CFStrings and passes them to a primary function: fn TranslateAndBuild

[23] FBtoC's Main changed: dictionary now created & sent when this code calls ProcessNotification()

[24] fn WriteErrorsFileForEditor now accepts CFURLRef input

[25] fn PrintErrorMessageToBuffer updated to use CFStrings and new fn MoveCFStringToRouteBuffer

[26] fn WriteErrorsFileForEditor update internally but still receives FSSpec input :-(

[27] All PRINT _toBuffer in fn PrintCompilerMessageToBuffer now uses a CFMutableString.

[28] The CFString in #27 is moved to the appropriate gFBBuffer via fn MoveCFStringToRouteBuffer which calls FB runtime FBCheckBuffer().

[29] Note: this approach relies on the existing ROUTE _toBuffer +(n)/ ROUTE _toScreen logic and doesn't require the new code to figure out which buffer is the current target of the ROUTE. Also note: Since buffer logic is mostly building and expanding a handle to allocated storage it doesn't impede 64-bit compiles.

[30] For later: update FBCheckBuffer() and associated code to use allocated pointers instead of allocated handles to avoid all the older SetHandlexxx calls.

[31] Fixed crash on a DisposePtr() in fn ReleaseSourceFileText caused by bug in fn TranslateFileRecursive

[32] More work on CompileFiles, CompileSource

[33] More work to replace gUnixPathToSourceFolder(pascal) with gCFUnixPathToSourceFolder(CF)

Right now fn SetUpFilePathsForThisProject uses CF internally but accepts FSSpecs and Str255. Next step is to have it accept CF input.

[34] fn CompileFiles converted to CFString usage and input

[35] fn CompileSource converted to CFString usage and input

[36] RUN command updated to accept:

- (a) a CFString literal
- (b) a CFString variable
- (c) CFURLRef variable

[37] FB Help for Run command updated to reflect new syntax shown in #36

[38] FB Help Appendix A: File object specifiers updated to recommend CFURLRefs and not FSRefs or FSSpecs.

[39] fn MakeDSym now accepts CFStrings

[40] Time display and calcs in fn TranslateAndBuild now CF instead of pascal/carbon

[41] fn CopyHelpBookFolderName now accepts, uses and returns CF objects

[42] 'kill resources' statement isn't supported and is ignored.

[43] fn AddConstantsFromResourcesToSymbolTable is ALMOST 64-bit clean but still passes a pascal string to AddNumericConstToSymbolTable()

[44] cleanup from 5.7.21. More direct use of fn CFStringFromPStr(pascalString) instead of letting returned value go stale before being used.

[45] from 5.7.16 - "all 'shutdown' and 'stop' code converted to call to 'ShowAlert' followed by [NSApp terminate:nil]"-- cleaned up.

[46] fn CopyUNIXPath activated as replacement for older FSSpec-based fn PathForUNIX.

[47] fn CopyObject now both accepts CFString input and processes CFStrings

[48] fn MakeDir now both accepts CFString input and processes CFStrings

[49] fn FileObjectExists now accepts CFStringRef input

[50] fn FileObjectExists reverted back to 'test -e' command because it is less code. Internally uses CFStrings.

[51] Editor's fn ReadTextFile and fn SaveTextFile updated to use new Cocoa ReadFile() and WriteFile(). Still returns a Handle and accepts an FSSpec.

[52] FBHelp's Files\$ doc corrected for options (2) Selecting a Folder and (3) For Selecting a File Name and Folder where a file may be Saved.

this was noted by Bernie and RC in a list discussion 18-June-2013.

[53] Replaced use of old '.rsrc' resources in editor because FBtoC doesn't process .rsrc files

- Editor window icons changed from cicon resources to pngs.
- Menu resources changed to toolbox menu calls (FB menu statements not used but possible for those converting from menu resource use)
- CFSwapInt16HostToBig() & CFSwapInt16BigToHost() was needed in 2008 but now removed for writing Editor Style Files.incl
- CFSwapInt16BigToHost() for reading existing version 2 style files remains.
- FB's Style file I/O using FB's OPEN, READ, WRITE, WRITE FILE, READ FILE changed to Foundation calls
- Minor update to SystemDirectoryCopyURL() to use signed 'domain' variable and quiet clang warning.
- fn CompilerPath, fn CopyUnixCommandFirstLineResponse and some others changed to accept CFString input

[54] fn CopyResources now accepts CFURL and CFStringRef input.

- Support for copying .rsrc old style resource fork files removed.
- FBtoC used to combine data from all resource fork files and create an AppName.rsrc with only a data fork. This is now gone
- Removed copying of 'Localized.rsrc' as it is no longer needed.
- 'rsrc' resource processing fns removed
- Some other misc small functions updated and added

[55] Editor incorrectly labeled text include files (with extension '.incl') as 'resources' inside data.fbproj (within projectname.fbproj)

- Happens only if the '.incl' file does not have a file type of 'TEXT'
- Building a project once cleans up the projectname.fbproj
- Editor code that decides whether to write 'include' vs. 'resources' in data.fbproj is fn WriteFileListToProjectDataFork()

[56] All 'shutdown' and 'stop' code converted to call to 'ShowAlert' followed by [NSApp terminate:nil];

[57] fn SendErrorMessageToLogAndEditor, fn SendInformativeMessageToLogAndEditor converted to CFStrings

[58] FBtoC's 'AppendToBuf()' upgraded to use CFString input. Used this approach (versus just loading a CFMutableString and passing it around) because all the AppendToBuf() code just uses pointers and doesn't invoke Carbon code. This way the logic is maintained and code can continue to call all the command line utilities(especially the compiler, linker, dsymutil, sed and strip). Other commands like 'rm' and 'mkdir' can be switched out for Cocoa but there is no pressing need for that. Tested with FBtoC/Editor source. Bernie's tests on some large projects also works fine.

[59] Lots of minor stuff just to clean up code, update comments and fix broken code.

Coding and implementation: Brian & Bernie

20150801 FB 5.7.14

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.13]

FB 5.7.14 requires OS X 10.6 or higher.

FB5 bug fixes:

- 'Appearance Window' inappropriately overwrote window attributes when the statement omitted window attributes.
- Editor and Project Manager window resizing was broken in OS X 10.6
- The hidden grow box change, implemented in 5.7.13, now hides grow boxes only for compositing windows.
- Turned off some debugs left on in prior release.

Enhancements:

- OSMajorMinor() returns both major and minor OS version in one call. Example: running in 10.7.5 it returns: major = 10 and minor = 7. Look for a usage example in a later list post.
- Tlxb MacWindows.incl header updated with drawer window and other helpful window calls:

```
// Drawer Window calls now recognized automatically in FB 5.7.14
toolbox fn GetDrawerPreferredEdge( WindowRef inDrawerWindow ) = OptionBits
toolbox fn SetDrawerPreferredEdge( WindowRef inDrawerWindow, OptionBits inEdge ) = OSStatus
toolbox fn GetDrawerCurrentEdge( WindowRef inDrawerWindow ) = OptionBits
toolbox fn GetDrawerState( WindowRef inDrawerWindow ) = WindowDrawerState
toolbox fn GetDrawerParent( WindowRef inDrawerWindow ) = WindowRef
toolbox fn SetDrawerParent( WindowRef inDrawerWindow, WindowRef inParent ) = OSStatus
toolbox fn SetDrawerOffsets( WindowRef inDrawerWindow, CGFloat inLeadingOffset, CGFloat
inTrailingOffset ) = OSStatus
toolbox fn GetDrawerOffsets( WindowRef inDrawerWindow, CGFloat * outLeadingOffset, CGFloat *
outTrailingOffset ) = OSStatus
toolbox fn ToggleDrawer( WindowRef inDrawerWindow ) = OSStatus
toolbox fn OpenDrawer( WindowRef inDrawerWindow, OptionBits inEdge, Boolean inAsync ) =
OSStatus
toolbox fn CloseDrawer( WindowRef inDrawerWindow, Boolean inAsync ) = OSStatus

// Helpful window-related calls now recogized automatically in FB 5.7.14
toolbox fn IsWindowActive( WindowRef inWindow ) = Boolean
toolbox fn ActivateWindow( WindowRef inWindow, Boolean inActivate ) = OSStatus
toolbox fn CopyWindowTitleAsCFString( WindowRef inWindow, CFStringRef * outString ) = OSStatus
toolbox fn HIWindowGetBounds( WindowRef inWindow, WindowRegionCode inRegion, HICoordinateSpace
inSpace, HIRect * outBounds ) = OSStatus
toolbox fn HIWindowSetBounds( WindowRef inWindow, WindowRegionCode inRegion, HICoordinateSpace
inSpace, const HIRect * inBounds ) = OSStatus
toolbox fn SetWindowResizeLimits( WindowRef inWindow, const HISize * inMinLimits, const HISize
* inMaxLimits ) = OSStatus
toolbox fn DetachSheetWindow( WindowRef inSheet ) = OSStatus
toolbox fn HIWindowSetToolbarView( WindowRef inWindow, HIViewRef inView ) = OSStatus
toolbox fn MacGetNextWindow( WindowRef window ) = WindowRef
```

Coding and implementation: Bernie, Brian

20150622 FB 5.7.13

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.12]

FB 5.7.13 requires OS X 10.6 or higher.

FB5 changes:

- System(_sysVers) changed to use modern Foundation calls instead of Gestalt.
 - N.B.: this call returns exactly what Apple's NSProcessInfo's operatingSystemVersionString provides. It concatenates the major, minor and bugfix version numbers

- and returns that number as a string: for OS X 10.9.4 major = 10, minor = 9, bugfix = 4, so 1094 would be returned as a SInt32. OS X 10.7.2 returns 1072, OSX 10.10.3 returns 10103 etc. Apple omits the bugfix version if it is zero, so 10.9.0 returns 109 and 10.10.0 returns 1010. This could impact code comparisons if the raw returned value is used as is.
- This change was prompted by Console log error messages like: "5/6/15 12:30:01.938 PM FutureBasic 5[1755]: WARNING: The Gestalt selector gestaltSystemVersion is returning 10.9.3 instead of 10.10.3. Use NSProcessInfo's operatingSystemVersion property to get correct system version number."
- Windows created with FB's 'Appearance Window' statement now omit the dated visible grow box.
- Garbled warnings for unused functions and unused forward declarations fixed
- 'dim as ptr p1, 2' caused crash. Fixed
- FB 5.7.13 source released to public.

Coding and implementation: Bernie, Brian, Michele

Notes: Items #3 and #4 are fixed in 5.7.13 re-released on 9-July-2015

20150219 FB 5.7.12

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.11]

FB 5.7.12 requires OS X 10.6 or higher.

FB5 changes:

- Many FBtoC source changes adopt Core Foundation and do specific replacements such as pascal string==>CFStringRef & FSSpec==>CFURLRefs.
 - in progress & mostly complete - UNIX source file fns are converted and some feeders and subordinates changed too.
 - in progress & mostly complete - ERROR HANDLING source file fns are either converted to Core Foundation or prepared for it
 - FSSpec replacement with CFURLRef/NSURL started (in progress, incomplete & not activated)
- Fixed: 'dim qs windowref wr' caused a crash.
- Fixed: 'File Not Found' error when a source file directory name contains a higher ASCII symbol (in this case 'f')
- Fixed: Typed pointer treated as untyped generates clang "invalid operands to binary expression" error when used in binary 'and'.

Coding and implementation: Bernie, Brian and Michele

Notes:

20141117 FB 5.7.11

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.10]

FB 5.7.11 requires OS X 10.6 or higher.

FB5 changes:

- FBtoC's log window rewritten in Cocoa nib and Objective-C
 - (a) Find bar added for searching the log after the build
- FBtoC's menus rewritten in Cocoa nib and Objective-C
- FB5's command menu now has "Analyze" and "Build" options.
 - (a) Analyze allows for easy one-off clang analyzes regardless of the Build Settings.
 - (b) Build omits the target app launch after the compile/link
 - (c) Superfluous "analyze" and "Launch after build" build settings options removed
- Tool tips added to the Build Settings window
- Added: Reminder warning "Compile as 'Objective-C'" is superfluous
- User is warned of a missing Quickdraw framework when choosing Build Settings' base SDK.
- Opening an old project is now smarter and checks if a previous path to a developer folder still exists
- Superfluous FBtoC option "Debug Last App in Xcode" removed.
- More general FBtoC code cleanup
- fn FD_SpecialDirectoryCreateCFURL now allows the programmer to check a returned OSStatus for an error

- Messaging from FBtoC to editor updated to use CFNotificationCenter instead of Apple Events
- NSLog updates:
 - (a) "Compile as Objective-C" removed from NSLog.incl
 - (b) NSLog window bounds now stored in user app's preferences instead of com.berniewylde.nslog.plist
 - (c) Requires 10.6+
- Fix: Path to Developer folder popup failed when more than one path present.
- Fix: Log options now appear only in FBtoC build settings window similar to version 5.7.8
- Fix: 'compilerVersion' now available for use again
- Fix: "Make Xcode without project Stripping" was unresponsive when selected immediately after an "analyze" build of the same code.
- Fix: SystemDirectoryCopyURL(), which supports the File Directory include, updated to support kPreferencesFolderType, kTemporaryFolderType plus a couple bug fixes.
- Fix: Editor incorrectly displays code block (if/else/endif, while/wend local fn/end fn) mismatching error in specific scenario(*). Visual problem only. It does compile.
- Fix: Find Again (CMD-G), after closing source window, find window and project window, opens unexpected "untitled" window and does a find. Should find nothing.
- Fix: Build Settings window fails to display in OS X 10.6.x
- Fix: More helpful editor error message when is unable to open project file. Thanks Michele.
- Workaround: when project with valid developer path is moved to another Mac, build crashes due to invalid path. A warning dialog is now displayed.

Coding and implementation: Bernie and Brian

Notes:

* A line of code exactly 35 characters long triggers a bug when code looks for '#' as the previous character. Turns out 35 is ASCII for the '#' char and comparison with the length byte of a pascal string causes the problem. Code changes for 'if' and 'else'

20141006 FB 5.7.10

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.9]

FB 5.7.10 requires OS X 10.6 or higher.

FB5 changes:

1. Cleanup from new Build Settings window implementation.
2. Cosmetic bug: log would show "FBtoC: build settings from project" when it was from FBtoC preferences. Right preferences being used but the log message was wrong.
3. New triple slash comment '///' allows FBer to select which comments to send to C/Objective-C generated code
4. FBtoC now generates NSBeep() for 64-bit builds
5. Bug fix: Any included resource with a plist file extension is now correctly put in /Contents/Resources and not /Contents *
6. Bug fix: Crash when Xcode and/or appropriate directories aren't available and Build Settings not consulted by user.
7. Added: Reminder warning "Compile as 'Objective-C'" is superfluous

Comments:

* the info.plist is correctly placed in /Contents

Coding and implementation by Bernie, Brian and Robert C.

20141003 FB 5.7.9

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.8]

FB 5.7.9 requires OS X 10.6 or higher. Please note the following changes before implementing 5.7.9.

FB5 changes:

1. The Build Settings window has been redesigned and re-written in Objective-C/Cocoa with a Cocoa nib with more options and flexibility.

- (a) clang is the compiler. Compiler selection is gone.
- (b) Formerly deprecated 'Allow dim a%, a&, a#, a\$' is now gone. Attempts to use will generate a "Redefined variable".
- (c) The Settings window does all SDK confirmation/coordination/existence checking. This eliminates build-time checking and also gives the programmer immediate SDK availability feedback.
- (d) Objective-C compile always used. No other option.
- (e) Intel architecture always used. PowerPC and Universal builds are gone and not supported
- (f) First build with PCH does not use anything in "Compiler Options". Subsequent builds use the PCH and any compiler options.
- (g) Clang analyze works. A bug reported on the list could not be reproduced but current clang analysis output looks normal.
- (h) Bug fix: "Min Deployment" all OS X releases in popup are selectable. Note: The SDK popup is still limited to those SDKs installed on the machine.
- (i) "-F/Library/Frameworks" now included automatically in compiler search paths *
- (j) Table views for 'Path to Developer folder' and 'Compiler options' enable the developer to save and quickly switch to other options without retyping.
- (k) New FB projects inherit current FBtoC Build Settings
- (l) Many FB Header, Editor and FBtoC source files were updated to squelch clang warnings about "illegal characters..."
All FB source compiles with no warnings.

Comments:

This is the first major FBtoC change in years. Bernie and I are excited the new 64-bit Build Settings coexists nicely with the older Carbon code.

FWIW: Overall code size decreased compared to the prior version.

Coding and implementation by Brian and Bernie.

* For some unknown reason, Xcode 5.1's clang sometimes doesn't find the framework when Build Settings specifies a 10.6 SDK. SDK 10.9 works ok. Clang from Xcode 6 works in either scenario. Bernie's testing was successful in all scenarios but mine wasn't.

20140703 FB 5.7.8

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.7]

FB 5.7.8 requires OS X 10.6 or higher.

FB5 changes:

Build Settings:

(1) For the Min OS deployment and Base SDK popup menus: Enabled menu items correspond to installed SDKs on the machine. Disabled menu items indicate SDKs not installed. For example, if the OS X 10.7 SDK is not installed, the 10.7 menu item will be visible but disabled.

(2) 'Path to Developer folder" defaults to: /Applications/Xcode.app/Contents/Developer

N.B. In order to use 10.7, 10.8 and 10.9 in these popups, users must make SDK changes to support QuickDraw headers that don't exist in the 10.7+ OS X releases. See the FBtoC [webpage](#) section titled "Optional QuickDraw headers support in OS X 10.7+" for more information.

New/Updated Headers/Runtime file(s):

NSLog.incl replaces the former FBLog.incl. See Bernie's [list post](#) for more NSLog information

New/Updated examples:

(1) FB_5_7_8_Examples/Miscellaneous/NSLog 1.2.8 -Updates and replaces the former FBLog

20140430 FB 5.7.7

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.6]

FB 5.7.7 requires OS X 10.6 or higher.

FB5 changes:

The 'Min OS deployment' and 'Base SDK' popups in Build Settings now support OS X 10.7, 10.8 and 10.9.

N.B. In order to use 10.7, 10.8 and 10.9 in these popups, users must make SDK changes to support QuickDraw headers that don't exist in the higher SDK releases. See the FBtoC [webpage](#) section titled "Optional QuickDraw headers support in OS X 10.7+" for more information.

New/Updated Headers/Runtime file(s):

Tlbg Aliases.incl: fn FSResolveAliasFile added

Tlbg HICocoaView.incl added

20121120 FB 5.7.6

<http://www.4toc.com/fb/index.htm> [you will need to reload the page if your browser cached the page mentioning FB 5.7.5]

FB 5.7.6 (like 5.7.5 and several releases before it) requires OS X 10.5 or higher.

FB5 changes:

718 Error Missing commas between function params not flagged

197 Error Malformed date/time in editor build log.

New/Updated Headers/Runtime file(s):

FBLog.incl: C printf commands are redirected to FBLog's window.

Minor FB Help updates for dialog function

New/Updated examples:

(1) FB_5_7_6_Examples/Miscellaneous/FBLog_and NSLog/printf test

Demonstrates the redirect of printf with FBLog.incl

(2)FB_5_7_6_Examples/Cocoa/FBNS

Demonstrates Cocoa calls in FB. Both .incl files and FB project version included.

(3)FB_5_7_6_Examples/Files/Bookmarks

Demonstrates bookmarks. Bookmarks are the successor to Alias Manager

(4)FB_5_7_6_Examples/Text/Scrolling text views/YAST folder

YAST(Yet Another Scrolling Text) uses Apple core code to build scrolling text fields.

Notes:

(a) Note the other scrolling text demos in YAST's parent directory

(b) Scrolling text demos in general are, in part, aimed at shortcomings in FB5's Scroll Button

(c) YAST uses Carbon's MLTE which is not recommended by Apple.

(5)FB_5_7_6_Examples/Cocoa/CocoaShell.bas

A basic Cocoa shell(no FB code but can run from FB)

Note: Steve Van Voorst has posted(on fbcocoa but some on the main list) a series of Cocoa/ Objective-C demos which are effectively a shell/template. Those exploring Cocoa should check out Steve's demos too.

(6) FB_5_7_6_Examples/Graphics/CoreGraphics (Quartz)/Quartz 1.12

20120918 FB 5.7.5

FB 5.7.5 (like 5.7, 5.7.1, 5.7.2, 5.7.3 and 5.7.4) requires OS X 10.5 or higher.

FB5 changes:

716 Error Util_EUTCKeyFilter.incl misidentified as a resource

717 Error Symbol table overflow. Table size increased by 8k

Please note: FB 5.7.4+ is required for use with OS X 10.8 (see bug #196)

New/Updated Headers/Runtime file(s):

Util_FileDirectoryII.incl:

This new include updates the runtime code for the special directory calls by replacing the now deprecated FSFindFolder() calls. The special directory (e.g. FD_SpecialDirectoryGetXXXXX) parameters have changed slightly. Please see the new example for details. The old include is still available

New/Updated examples:

FB_5_7_5_Examples/Files/File_Directory examples/FileDirectory-SpecDirsUpdated.bas

20120802 FB 5.7.4

FB 5.7.4 (like 5.7, 5.7.1, 5.7.2 and 5.7.3) requires OS X 10.5 or higher.

Please note: FB 5.7.4 is required for use with OS X 10.8 (see bug #196)

FB5 changes:

196 Crash Clicking on project window icons causes crash in OS X 10.8

713 Error Appending pascal string in 'end fn = ' fails

714 Error Error message for non-terminated '#if' is poor

20120528 FB 5.7.3

FB 5.7.3 (like 5.7, 5.7.1 and 5.7.2) requires OS X 10.5 or higher.

FBtoC changes:

704 Error Changes in FBtoC.h not propagated for clang+pch builds

706 Cleanup When save button clicked, fix common user mistakes for 'Path to Developer folder'

708 Error Long variable(symbol) names generate translation and compiler errors

709 Error Make Xcode Project -> Warning: '/' in block comment

710 Cleanup Remove '_A' suffix from translated array names

711 Error Using 'double' variable type with Input fails in FB 5.7.2+

712 Error Cursor not changing from arrow to I-Beam and back in some cases.

New/Updated Headers file:

New - FBLog.incl

Updated - Util_EUTCKeyFilter.incl

Updated - Util_CE.incl

New examples:

FB_5_7_Examples/Miscellaneous/FBLog_and NSLog/NSLog Demo

20120310 FB 5.7.2

FB 5.7.2 (like 5.7 and 5.7.1) requires OS X 10.5 or higher.

Editor changes:

191 Feature Compiler warnings/errors jump to FB code

FBtoC changes:

670 Nicety C files should have UNIX LF line endings only, not mixed with CR

690 Feature Provide FB line info with compiler warnings/errors

691 Feature 'Unused function' warning should jump to definition

692 Error Window title corrupted

695 Feature dialog(-1) returns wndNum for all window events [including _preview]

696 Nicety 'Debug Last App' should attempt to open Xcode 3

701 Cleanup -DDECARBONATE should require -m64

702* Feature Support Xcode 4.3

703 Error Frozen menus from 'window output 0' in DoDialog

Updated Headers file:

Tlbb CFByteOrder.incl

New examples:

FB_5_7_Examples/Graphics/CoreGraphics (Quartz)/CGGradient_demo

FB_5_7_Examples/Graphics/CoreGraphics (Quartz)/RoundRect+CGGradient

FB_5_7_Examples/Graphics/CoreGraphics (Quartz)/UserPaneGrids

* Xcode 4.3 is installed in the Applications folder. To use with FB 5.7.2, Settings > Advanced must indicate the new location.

Path to Developer folder: /Applications/Xcode.app/Contents/Developer

The checkbox 'Use compiler from developer folder' checkbox should be ON.

20120120 FB 5.7.1

FB 5.7.1 (like its predecessor 5.7) requires OS X 10.5 or higher.

Editor changes:

184	Cosmetic	Source code drawn in project window
185	Error	void functions don't show in function pop-ups
186	Cleanup	Bad horiz scrolling of background window by mouse wheel

FBtoC changes:

343	Error	gcc error from pointer.offsetConst\$ arg to toolbox Str255 param
505	Error	internal error from dim a as ^Point, b&
558	Error	Compiler error from block comment in BeginC...EndC
566	Error	Comment ending in opt-L eats next line of code
617	Error	inc(p) and p++ are different if p is ^Type
619	Error	Make Xcode Project strips main() from command-line tool
666	Error	'dim as ..FooType bar' is pointer but should be Handle
667	Error	Can't cast address arguments freely and correctly
669	Error	Repeated 'clear local' gives internal fatal error
672	Error	Build error 'no include path in which to search for Carbon/Carbon.h'
673	Error	Default compiler in OS X 10.7 should be clang
674	Error	'void clear local fn Foo as long' should be error
675	Feature	Allow spaces in path to Developer folder
676	Error	Can't cast to (float)
677	Error	Record assignment wrongly disallowed
678	Error	Return value from function of type float treated as integer
679	Error	'not(a binop b)' wrongly translated
680	Feature	Maximum number of dynamic arrays doubled to 64
681	Error	Allow record field as file specifier in 'open' statement
682	Feature	Add 'Crash' button to all error alerts
683	Error	Buffer overflow when linking huge project
684	Error	Generate Localizable Strings fails when compiled as Objective-C
686	Error	Unsigned function type wrongly disallowed
687	Error	Crash from invalid syntax for 'button', 'appearance button' and 'appearance window'
688	Error	FD_SpecialDirectoryXxxx() should create directory if absent

Note:

Projects and standalone builds that used clang with the precompiled header option will encounter an error when first built with FB 5.7.1

Error: clang:

redefinition of 'gFBDynArrayInfo' with a different type [line 115:14 of...]

The simplest workaround is to trash the build_temp folder. An alternative is to switch temporarily to another compiler.

New Examples:

FB_5_7_Examples/Graphics/CoreGraphics (Quartz)/SelfCenteringHIImageView

20111115 FB 5.7

FB 5.7 requires OS X 10.5 or higher.

Editor changes:

140	Error	Tab key moves focus to toolbar and prevents subsequent editor window typing
142	Error	'REM' slider in document window's toolbar misbehaves
162	Error	Wrong indentation from opt-space between 'end' and 'fn select record...'

174	Error	Edit menu first item is always 'Can't Undo'
088	Feature	Allow mouse wheel/trackpad scrolling of background window
177	Feature	Headers ('H') button in doc window's toolbar
179	Feature	Adding file to project should make copy if not in project folder
180	Feature	Project menu item: New Project File...
181	Feature	Help menu item: Search Apple Headers.app

FBtoC changes:

656	Error	Hang from missing right parenthesis in 's\$ = edit\$(1'
650	Cleanup	Deprecate type-identifier suffices (except '\$') on function names
651	Cleanup	Deprecate same name for scalar and array variable: 'dim as long a, a(100)'
652	Cleanup	Deprecate vars with same base-name but different type-identifier suffices: 'dim a%, a#, a\$...'
653	Cleanup	Deprecate FB 16-bit 'int' type
657	Cleanup	Passing pointer instead of record argument should be warning
654	Feature	New keyword 'float' as synonym for 'single'
658	Feature	Allow pass-by-value records in user-defined functions
665	Feature	Allow C's dereferencing syntax for typed pointers
647*	Feature	New 'if.../else.../endif' syntax
671	Nicety	Add Bundle identifier to default Info.plist

New Headers file:

Tlhx CFURLEnumerator.incl

Updated Headers files:

Tlhx CFURL.incl

Tlhx Controls.incl

Tlhx MacWindows.incl

Util_Containers.incl

New functions in Console runtime:

```
fn ConsolePrintCType( typeRef as CTypeRef ) // any CType
fn ConsolePrintCString( string as CStringRef ) // CString only; fast
```

New examples:

FB_5_7_Examples/Files/CFURLEnumerator

FB_5_7_Examples/Files/NSOpenPanel & NSSavePanel

FB_5_7_Examples/Files/Search Apple Headers

FB_5_7_Examples/Graphics/CoreGraphics (Quartz)/ImageEdit skeleton

* The new syntax:

```
if ( 1 == 1 )
    beep
else
    stop
end if
```

is equivalent to the traditional (and still supported) syntax:

```
long if ( 1 == 1 )
    beep
xelse
    stop
end if
```

20110912 FB 5.6.2

Major bug-fixes:

641	Error	Crash from DynamicNextElement()
642	Error	DynamicRemoveItems() attempts to remove non-existent items

Feature introducing type safety to FB:

621	Feature	Allow compiler to warn of some type-mismatched pointer assignments
-----	---------	--

Cleaning up FB:

636 Cleanup Deprecate shorthands '& % |' for poke
637 Cleanup Deprecate 'enterproc ... exitproc'
638 Cleanup Disallow 'def' and 'usr' function declarations and calls

Improvements in translated C code:

622 Cosmetic Better translation of 'if (pointerVar) ...'
627 Cosmetic Pointers to scalar types should not be void*
628 Cosmetic Fewer parentheses in expressions with bitwise operators
634 Cosmetic Translate '++' as such, not '+= 1'
640 Cosmetic Translate 'rec = ptrVar' as 'rec = *ptrVar;'

Feature allowing improvement in translated C code:

635 Cosmetic Void keyword for functions ('void local fn Foo...')

Minor bug-fixes and other changes:

620 Error Function type mistranslated when ^Type or ^^Type
625 Feature Allow 'include "BarFolder:foo.c"'
626 Error Arguments to variadic functions should not be cast
639 Feature Use of 'fn ToolboxProcedure' as rval should be error
645 Cleanup Warn against fn CFSTR(stringVar)

New example:

FB_5_6_2_Examples/Controls/Custom text browser

20110827 FB 5.6.1

FBtoC general changes:

599 Feature Optional translation of _constants to symbols, not magic numbers
600 Feature New directive: compile [as] "Objective-C"
601 Feature New 'select switch' statement
603 Feature NULL keyword
615* Nicety Warn of long var dereferenced as though pointer
616† Nicety Warn of implied addition of _constants

* Until now, longs and pointers were treated almost interchangeably.

```
local fn bar( @foo& )  
    foo.nil& = 0  
end fn
```

This abuse is now deprecated.

⚠ Warning: long var 'foo' used as pointer in line 4 of ...

† The weird syntax '_foobar = _foo_bar' is now deprecated.

⚠ Warning: implied addition of constants; suggested form: '_foo + _bar' in ...

FBtoC changes relating to Make Xcode Project:

604 Error Resource files copied as empty data-fork by Make Xcode Project
607 Feature Warn of 'include library' framework not passed to Xcode project
609 Error Xib files wrongly converted to nib by Make Xcode Project
611 Feature Warn of library linker commands not passed to Xcode project
613 Error Nibs give ibtool error in Xcode 4 after Make Xcode Project

Updated Headers files:

Tlxb CarbonEvents.incl [fixes RegisterEventHotKey()]
Tlxb CFCharacterSet.incl [fixes CFCharacterSetXxxx constants]
Others too numerous to list.

New examples:

FB_5_6_1_Examples/Menus/CFStatusItem
FB_5_6_1_Examples/Miscellaneous/CalCalendar

Updated examples:

Too numerous to list.

Deleted example:

FB_5_6_x_Examples/Text/SpellCheck [crashes mysteriously]

20110722 FB 5.6

Editor feature:

New menu item Tools > Open Headers Folder

FBtoC bug-fixes and features:

591	Nicety	Default compiler should be gcc 4.2
592	Nicety	Show compiler path in error message when it cannot be found
593	Cosmetic	Increase ConsoleWindow's height and font size
594	Feature	Complex arithmetic, variables and functions
595	Error	Wrong formatting of signed zero by 'using'
597	Cosmetic	Change 'Max OS features' to 'Base SDK', matching Xcode
598	Error	Crash after multiple 'input' in ConsoleWindow

New Headers files:

Tlbg CFCalendar.incl

Tlbg complex.incl

Tlbg MDItem.incl

Updated Headers files:

Tlbg CoreGraphics.incl now has CGDisplayFade functions

Util_CFPrefsCFStrings.incl now has CFPrefsGet/SetCGRect()

New examples:

FB_5_6_Examples/Calculations/complex arithmetic

FB_5_6_Examples/Cocoa/CocoaControls_noNIB

FB_5_6_Examples/Files/Metadata demo

FB_5_6_Examples/Miscellaneous/CFCalendar demo

FB_5_6_Examples/Miscellaneous/Fade screen to black

FB_5_6_Examples/Text/Print Columnar Data

Updated example:

FB_5_6_Examples/CoreFoundation/CFArray of Dicts Proj

FB_5_6_Examples/Miscellaneous/FBLog

Editor feature:

New menu item Tools > Open Headers Folder

FBtoC bug-fixes and features:

591	Nicety	Default compiler should be gcc 4.2
592	Nicety	Show compiler path in error message when it cannot be found
593	Cosmetic	Increase ConsoleWindow's height and font size
594	Feature	Complex arithmetic, variables and functions
595	Error	Wrong formatting of signed zero by 'using'
597	Cosmetic	Change 'Max OS features' to 'Base SDK', matching Xcode
598	Error	Crash after multiple 'input' in ConsoleWindow

New Headers files:

Tlbg CFCalendar.incl

Tlbg complex.incl

Tlbg MDItem.incl

Updated Headers files:

Tlbg CoreGraphics.incl now has CGDisplayFade functions

Util_CFPrefsCFStrings.incl now has CFPrefsGet/SetCGRect()

New examples:

FB_5_6_Examples/Calculations/complex arithmetic

FB_5_6_Examples/Cocoa/CocoaControls_noNIB

FB_5_6_Examples/Files/Metadata demo

FB_5_6_Examples/Miscellaneous/CFCalendar demo

FB_5_6_Examples/Miscellaneous/Fade screen to black

FB_5_6_Examples/Text/Print Columnar Data

Updated example:

FB_5_6_Examples/CoreFoundation/CFArray of Dicts Proj

20110421 FB 5.5.1

Editor bug-fix:

167 Error Strange beep on double-clicking symbol preceded by option-space

FBtoC bug-fixes and features:

584 Error Freeze with 'Wrong number of arguments' error
585 Nicety More compact representation of 'data' strings
586 Error Crash on multi-line strings > 255 chars
588 Feature Advanced settings in drawer
588 Feature 'system long' should allow comma-separated variable list

Updated header:

Tlhx CoreGraphics.incl now includes CGImageSource functions

New examples:

FB_5_5_1_Examples/Cocoa/NoNib_FBtoCocoa
FB_5_5_1_Examples/Cocoa/NSTableView_FBtoCocoa
FB_5_5_1_Examples/Controls/bmpImage_btn/
FB_5_5_1_Examples/Controls/HICustomView
FB_5_5_1_Examples/Controls/ThumbnailButton/
FB_5_5_1_Examples/Games/Connect4/
FB_5_5_1_Examples/Graphics/CoreGraphics (Quartz)/CGPathContainsPoint
FB_5_5_1_Examples/Graphics/CoreGraphics (Quartz)/Floodfill
FB_5_5_1_Examples/Graphics/CoreGraphics (Quartz)/ImageResizer
FB_5_5_1_Examples/Text/CoreText styled text
FB_5_5_1_Examples/Text/CoreText with CGColor

Updated example:

FB_5_5_1_Examples/Games/Spandrel_1_4/

20110122 FB 5.5

Major changes:

[1] FutureBasic 5.app is now a self-contained IDE; its supporting file objects (FBtoC.app and the folders build_goodies and editor_support) are inside the app bundle.

[2] An Xcode project is created in the project source folder instead of build_temp as previously. This change makes it less liable to inadvertent deletion. The Xcode project is contained in a folder named <project_name>_XcodeProject.

[3] The User Libraries folder has moved to ~/Library/Application Support/FutureBasic/, alongside Scripts and Style Files.

FBtoC bug-fixes and features:

573 Error Hangs when parsing conditional with @nonexistent_var
574 Feature Make Xcode project in source folder, not build_temp
575 Error FlushWindowBuffer doesn't
576 Error apndstr() wrongly byte-swaps the count field
578 Error val() and val&() return wrong value for 16-digit hex strings
579 Error 'read' gets wrong value for 16-digit hex numbers in 'data' statement
581 Error Link error from Xcode project containing Objective-C code
582 Error NavDialog() releases saveFileName and message wrongly

New examples:

FB_5_5_Examples/Cocoa/DotView_FBtoCocoa
FB_5_5_Examples/Cocoa/GaussianBlur_demo
FB_5_5_Examples/Cocoa/SndPlayer_FBCocoa
FB_5_5_Examples/Controls/HISegmentedView
FB_5_5_Examples/Graphics/CoreGraphics (Quartz)/HIImageView_swipe
FB_5_5_Examples/Text/Programmatic HITextView

Updated example:

20101116 FB 5.4.8 and FBtoC 1.4.8

Editor bug-fix:

A long-standing 'dangling handle' bug has been found, that could cause a variety of crashes.

FBtoC bug fixes and new features:

570 Error Resources not passed to Xcode project
571 Error Make Xcode Project gives error 'FBtoC_Placeholder_RETURN' undeclared
572 Cleanup Code inflation by numerous gosub/returns*

New example:

FB_5_4_8_Examples/Controls/BrowserCustomItem

*The new implementation of gosub/return does not support optimized compilation. If your code uses gosub/return and needs optimization, you will have to replace every subroutine by an ordinary local fn.

20101008 FB 5.4.7 and FBtoC 1.4.7

FBtoC bug fixes and new features:

536 Cleanup Redundant files copied to XcodeProject
556 Cleanup Editor Errors window garbles multiple errors/comments on same line
557 Error gosub/return broken
559 Error route _toBuffer does not work with ConsoleWindow
560 Feature Support llvm-gcc 4.2 compiler
561 Error _controlkey constant missing
563 Feature Preference setting to control source-file combining

New Headers file:

Util_NibCFStrings.incl

New example:

FB_5_4_7_Examples/Sound/Record_sound_with_QuickTime

Updated examples:

FB_5_4_7_Examples/Games/Spandrel_1_2 (chess)
FB_5_4_7_Examples/Sound/Play Sound file

20100707 FB 5.4.6 and FBtoC 1.4.6

Editor changes:

Two new features mean that you will no longer lose your scripts and style files on updating to new versions of FutureBasic 5.
[1] Scripts now reside in ~/Library/Application Support/FutureBasic/Scripts. When the editor starts it looks for this folder. If the folder does not exist, it is created and the sample scripts are copied from the current editor_support folder.
[2] Style files now reside in ~/Library/Application Support/FutureBasic/Styles. When the editor starts it looks for this folder. If the folder does not exist, it is created and the style files are copied from the current editor_support folder.

FutureBasic 5 Help:

This now includes most of the material formerly contained in the FBtoC Help Book. The latter is largely obsolete.

FBtoC bug fixes and new features:

400 Error preprocessor bug for comments with quote char
463 Feature pass comments on to C source
519 Error Misleading error for missing project
522 Feature Strip unused FB runtime and C runtime functions when making Xcode project
527 Nicety Cleaner translation of simple 'if' conditions
535 Nicety Make Xcode Project should set optimization -O0
538 Feature Send runtime errors to console.log instead of showing fatal alert

```

539 Cleanup    Compiler warning from A$ = "(?)"
540 Cleanup    Warnings from Subs Quick Event Loop.Incl
541 Error       write# stringVar gives strange results
542 Error       'line input #' should terminate on <lf> as well as <cr>
543 Feature     Allow casting e.g. x = (double)y
544 Error       instr() treats chr$(0) and chr$(1) alike
545 Feature     Allow 64-bit integer literals such as 1ULL, -1LL
548 Error       'Warn of unused functions' sometimes didn't
549 Error       FSCatalogInfo.permission wrongly defined in Headers
550 Error       LP64 types should have size 8 in static expressions
553 Error       inkey$ for command line tool not implemented
554 Error       'Use precompiled header' breaks with clang 1.5
555 Feature     Allow static string expression (_foo$ + "bar") as 'output' filename

```

New Headers file:

```
Tlbx CFNumberFormatter.incl
```

New examples:

```

FB_5_4_6_Examples/Controls/CustomShapedControl
FB_5_4_6_Examples/Controls/CustomDrawnControl (Quartz)
FB_5_4_6_Examples/CoreFoundation/PrintCFString
FB_5_4_6_Examples/Files/FileObjectPermissions

```

Updated examples:

```

FB_5_4_6_Examples/AppleScript/LaunchMailPressSendII
FB_5_4_6_Examples/Controls/CustomDrawnControl (QuickDraw)
FB_5_4_6_Examples/Files/Droplet apps/Accept folder drop
FB_5_4_6_Examples/Files/Droplet apps/FinderInfo
FB_5_4_6_Examples/Files/Droplet apps/FinderInfo replacement
FB_5_4_6_Examples/Games/Spandrel 1.1 (chess)

```

20100406 FB 5.4.5 and FBtoC 1.4.5

Editor changes:

```

[1] 'Find in Next File' item (Command-option-G) in Edit menu.
[2] A crash from typing '#if def _someLongConstant' is fixed.

```

FBtoC bug fixes and new features:

```

510 Error      edit$( without closing parenthesis crashes FBtoC
516 Error      Dim syntax error not flagged
521 Error      Function declared as Str63 is unusable
522 Feature     Strip unused FB runtime and C runtime functions during Make Xcode Project
523 Nicety      Insert spaces around '*' in C translation of typed pointer parameters
524 Nicety      Symbolise boolean constants in C translation
525 Nicety      C translation should have NULL instead of (void*)(0)
526 Cleanup     Remove trigraphs and their support
528 # Feature   Option to build command line tool instead of app
530 # Feature   Allow 64-bit compilation of command line tool
531 Feature     Bounds checking should work for record fields that are arrays
532 Error       Hang when parsing call to undefined function
533 Feature     Allow dereference of pointer that is a record field and a pointer to another
record
537 Cleanup     Remove thousands of obsolete predefined constants

```

see the 'tool' example below. Also see FB5 Help: Appendix J - Command Line Tools.

Function change:

```

GetFormattedDateTimeAsString() has been replaced by DateFormatCreateStringWithAbsoluteTime().
Note the difference in parameter order:
//toolbox fn GetFormattedDateTimeAsString( CFAbsoluteTime at, CFStringRef formatString ) =
CFStringRef
toolbox fn DateFormatCreateStringWithAbsoluteTime( CFStringRef formatString, CFAbsoluteTime
at ) = CFStringRef

```


Revised Headers file:

Tlbbx HIView.incl [fixes definition of HIViewContentInfo]

New examples:

FB_5_4_5_Examples/Command Line Tool/tool

FB_5_4_5_Examples/CoreFoundation/CFString left, right, mid

FB_5_4_5_Examples/CoreFoundation/CFArray of Dicts Proj

FB_5_4_5_Examples/CoreFoundation/CFStrings in FB console

FB_5_4_5_Examples/Dialogs and Windows/ImageView as window background

FB_5_4_5_Examples/Miscellaneous/CFDateFormatter test

Updated examples:

FB_5_4_5_Examples/Miscellaneous/DateExample

FB_5_4_5_Examples/CFPrefs Demos/CF string version/Prefs w/Nib & CE

FB_5_4_5_Examples/CoreFoundation/CFAdditions

FB_5_4_5_Examples/Games/Spandrel 1.0 (chess) [replaces RChess 3]

20100214 FB 5.4.4 and FBtoC 1.4.4

Editor changes:

[1] Document windows are now live-resizing.

[2] The text of items in the Errors window can now be copied with Command-C.

[3] A text selection bug is fixed. It occurred when the mouse was dragged into the left margin.
(Thanks to Robert C.)

[4] Navigation arrows in the Help Book now work as expected.

FBtoC bug fixes:

414 Nicety 'Make Xcode Project' leaves stray .proj in build/NewProjectTemplate/

513 Error Mouse clicks in compositing window with FB Events and 'on mouse' vector not detected

514 Error Orphan 'case' statement not identified as error

515 Error Xcode project template not deleted after Xcode project created

517 Error mki\$ function can return wrong values

518 Error edit\$(ef) crashes if the field has not been created

520 Error Linker warning shows as pseudo error

New date\$ and time\$ functions:

These are now implemented via Core Foundation. They can optionally retrieve both the date and time in one call, and format it in almost any conceivably useful way. See FutureBasic 5 Help for details. The runtime code for date\$/time\$ may be called directly to format any date/time (not just the current one). See DateExample below.

New examples:

FB_5_4_4_Examples/Miscellaneous/DateExample

FB_5_4_4_Examples/Miscellaneous/DockTileBadge

FB_5_4_4_Examples/Files/EnumerateDirectory

FB_5_4_4_Examples/Files/Alias demo

FB_5_4_4_Examples/Text/CFAdditions

FB_5_4_4_Examples/Text/CFAdditions/CFScanner

Updated examples:

FB_5_4_4_Examples/Nib to FB wndNum & btnNum

FB_5_4_4_Examples/App with icon

FB_5_4_4_Examples/Games/RChess_3.0 [replacing RChess_3.0a2]

20091221 FB 5.4.3 and FBtoC 1.4.3

Important Changes for 5.4.3 (note: minor bug fixes and other internal fixes are not shown):

<u>Type</u>	<u>Description</u>
-------------	--------------------

New	new xib format support added. This is the newer XML nib used in Xcode.
-----	--

New	BeginCCode, BeginCFunction, BeginCDeclaration and EndC are improved methods for including C source directly..
-----	---

Many examples & documentation updated to use the new BeginCCode BeginCFunction BeginDeclaration keywords.

New .c, .h, .m files are now selectable in "Add to project" file dialog.
New _FBConsoleWndNum constant allows Console window customization .
The user may modify the default window in useful ways, for instance:
include "ConsoleWindow"
window _FBConsoleWndNum, "My tall debug stuff", (5,47)-(500, system(_scrnHeight) -
5)
See new console application demo in Examples/Games

Fix open "A" and open "R" now create the file if it did not exist.
Fix Menu items ending with '!' or '^' or '<' or '/' no longer crash the app.
Fix Linker errors are now passed to editor

New examples:

FB_5_4_3_Examples > CarbonEvents > EventMonitorTarget
FB_5_4_3_Examples > Games > RChess_3.0a2

20091107 FB 5.4.2 and FBtoC 1.4.2

This release is being made early, because of the severity of bug #506 (introduced in 5.4.1).

Editor is unchanged.

Bug-fixes and minor features in FBtoC 1.4.2:

323	Error	input "prompt"; x does not work in console mode
422	Error	'dim as long a\$' should be an error
467	Error	gCntr = edit\$(1) should work for > 255 chars
471	Error	missing '"' gives bad C code
484	Error	bad C code from @"\"
493	Error	str\$("should be translation error")
498	Nicety	include "ConsoleWindow" (alternative to include "FBRuntime.incl")
499	Error	str\$(fpNum) and 'print fpNum' give different results
500	Error	prevent clang Universal build with <10.5 deployment
501	Error	in Console Window, 'print fpNum' ignores gFBFloatMaxDigits
502	Error	can't access 3D array in record
503	Error	prevent array declaration with >4 dimensions
506	Error	can't type upper-case letters in edit fields

New examples:

FB_5_4_2_Examples > Calculations > fast sin() and cos()
FB_5_4_2_Examples > Calculations > Expression Evaluator
FB_5_4_2_Examples > Calculations > Measurement of code performance

20091021 FB 5.4.1 and FBtoC 1.4.1

Main changes in FB 5.4.1

[1] Warnings for a common mistake: forgetting to assign a return value (end fn = someValue) to a function whose value is used elsewhere in the program.

```
local fn bar
dim as OSStatus err
err = _paramErr
end fn // oops, should have '= err'
```

```
dim as OSStatus err
err = fn bar
if ( err != _noErr ) then stop "bar() failed" // oops, never get here
```

?? Warning: Returned value of bar() is always 0 in line 9 of test.bas
?? Warning: Function should have explicit return value: bar()

[2] Choice of compiler. Build settings window has a pop-up menu with four options.
- gcc 4.0 is the default.

- gcc 4.2 is available on 10.5 and 10.6. It cannot build apps to run on 10.3 or 10.4.
- clang is available on 10.6 only. Very fast but somewhat flaky. Best used for development only (change to gcc 4.0 for release build).
- clang (analysis) performs static analysis of your code, giving interesting warnings; nothing is actually compiled.

Minor changes in Editor 5.4.1:

- [1] The menu item Project > Remove Item has been deleted.
- [2] Bug-fixes in the following areas: text drawing, menus, Find/Replace window.

Bug-fixes and minor features in FBtoC 1.4.1:

```

476     edit field with filter fn receives Cmd-chars
486     default MinOS deployment 10.4; remove 10.2 target from UI
490     >1-dimension array parameter wrongly translated
492     warn of unused fn prototypes as well as unused fns
494     allow fn instead of def and usr function calls
495     deprecate usr and def function calls
496     terabyte file access; lof(), loc() and rec() return SInt64
497     implement open "N" to give shared access

```

New examples:

```

FB_5_4_1_Examples > Graphics > QuickDraw > Image dithering
FB_5_4_1_Examples > Graphics > QuickDraw > Ray Tracing
FB_5_4_1_Examples > Graphics > CoreGraphics > CGGlyphs

```

20090729 FutureBasic 5.3 and FBtoC 1.3

Issues arising during beta testing of the editor have been resolved. Thanks to all who reported bugs.

Notable changes from FB 5.2:

1. Universal binary editor runs native on Intel.
2. Editor's Build Log shows progress of translation and subsequent build steps.
3. Editor doc windows are useable when toolbar hidden.
4. Editor has Help Book that in part describes FB5, not FB4.
5. New icons for editor, FBtoC, and FB documents (log out/in to see).
6. Find/Change window is resizeable.
7. Source code text is drawn with the casing in which it was typed.
8. New projects have extension .fbproj.

Note that FutureBasic 5.3 editor requires FBtoC 1.3 and vice versa; you can't mix and match with earlier versions.

Bug fixes and minor features in FBtoC:

```

421     Error      setting gFBQuit true in DoDialog() doesn't quit
443     Error      unhelpful error msg from malformed _constant
444     Error      string[pointer] should be translation error
445     Error      *.m files not copied during Make Xcode Project
446     Error      string stack bug from 'input z'
447     Feature     'Hide on suspend' preference checkbox
448     Error      Objective-C source files become *.m.m
450     Error      flag missing commas in argument list
452     Error      'Bad case statement' error could force recompilation
453     Feature     allow constants to start with numeral
455     Error      nested index with 'Check array bounds' gives various bugs
456     Feature     change 'Use DWARF' radio button to 'Put line # in app'
457     Error      .c, .m and .h files could open in FB when double-clicked
458     Error      edit$( btnNum ) returns nothing if text len > 255
459     Feature     Save Log As...
460     Nicety      'More warnings' should not include -Wunused-parameter
462     Feature     @"some
string" --> CFSTR( "some
string" )
464     Error      output window wrong for _btnClick in floating window
470     Nicety      explicit error for protoype illegally in local fn

```

New Headers files:

```
Util_CFUN.incl
Util_Nib.incl
```

New examples:

FB_5_3_Examples > Carbon in Cocoa

FB_5_3_Examples > Graphics > CoreGraphics (Quartz) > AceOfHearts, ImageMask and ColorMask

FB_5_3_Examples > Files > Accept folder drop

Some old kludgy declarations in Toolbox CFDate.incl have changed to match the official Apple declarations:

```
CFGregorianCalendarIsValid()
CFGregorianCalendarGetAbsoluteTime()
CFAbsoluteTimeGetGregorianCalendar()
CFAbsoluteTimeAddGregorianCalendarUnits()
CFAbsoluteTimeGetDifferenceAsGregorianCalendarUnits()
```

20090625 Beta version (beta 03) of Universal FB Editor Available

-----Overview-----

The FBtoC team is pleased to announce beta testing for a new universal binary version of the FB Editor. The beta editor is a conversion of the older editor (included with the 'FB_5_2_Package') and thus retains most of the old editor's basic design and functionality. Many of the changes are internal to facilitate a universal binary build but some features were upgraded. Please note: Even though there are many proposed (but not implemented yet) upgrades (many internal) to this new UB editor, the team believes putting a Universal editor in the hands of the FB community is a higher priority than waiting for a 'finished' product. Even though it seems unlikely Apple will remove support of Rosetta in OS X 10.6, they have yet to clarify if it will be included, so having a universal binary editor protects the FB community for now.

-----Changes-----

- (1) Feature: Look up of Apple toolbox calls in Xcode Docs by highlighting the call and using either the contextual menu or the help menu to initiate.
- (2) Bug fix: High level project level disclosure arrow (flipper) showed only "File" label when project closed & re-opened
- (3) Bug fix: Warnings and errors from gcc are mishandled by showDbClickedError() as reported by Andy P. and RC.

-----Installation-----

- (1) Unzip the executable which is named 'FutureBasic_5_3beta' ---do NOT rename
- (2) Put it in the same folder(which is typically 'FB_5_2_Package' for the current release of FB5) as your current 'FutureBasic 5' application.
- (3) Launch 'FutureBasic_5_3beta' instead of 'FutureBasic 5'

Notes:

- (a) The beta editor requires FB5 and does not support FB4 or older FB releases
- (b) The beta editor is built with debugging symbols turned on to provide more debugging information in the event of a failure. The included debugging symbols increase the size of the executable but should not otherwise impact operation.

20090619 Beta version 02 of Universal FB Editor Available

-----Overview-----

The FBtoC team is pleased to announce beta testing for a new universal binary version of the FB Editor. The beta editor is a conversion of the older editor (included with the 'FB_5_2_Package') and thus retains most of the old editor's basic design and functionality. Many of the changes are internal to facilitate a universal binary build but some features were upgraded. Please note: Even though there are many proposed (but not implemented yet) upgrades (many internal) to this new UB editor, the team believes putting a Universal editor in the hands of the FB community is a higher priority than waiting for a 'finished' product. Even though it seems unlikely Apple will remove support of Rosetta in OS X 10.6, they have yet to clarify if it will be included, so having a universal binary editor protects the FB community for now.

-----Fixes for Beta 02-----

- [1] Crash at startup on PowerPC Mac in fn AddToTLBXList

The crash was the primary reason for releasing another beta so soon after the first beta.

[2] Compile info window doesn't show *.c files
[3] Match compile info format from FB 5.2

[4] Various changes to help menu to reinstate "Apple Web Reference" & other functionality
It is now possible to highlight an Apple toolbox call in the editor window and do a find on Apple's developer site by selecting "Apple Web Reference"

[5] Wrong fn name at top of editor window
[6] Initialize fn name and line info fields on doc open
[7] Re-opening a project should make the PM window front
[8] Help menu should allow access to miscellaneous files
[9] horiz scroll bug
[10] Compile Info window elides some <cr>s

-----Installation-----

(1) Unzip the executable which is named 'FutureBasic_5_3beta' ---do NOT rename
(2) Put it in the same folder(which is typically 'FB_5_2_Package' for the current release of FB5) as your current 'FutureBasic 5' application.
(3) Launch 'FutureBasic_5_3beta' instead of 'FutureBasic 5'

Notes:

(a) The beta editor requires FB5 and does not support FB4 or older FB releases
(b) The beta editor is built with debugging symbols turned on to provide more debugging information in the event of a failure. The included debugging symbols increase the size of the executable but should not otherwise impact operation.

-----Reporting Bugs-----

Post to the FB list with the subject: 'FB Editor: yourBug'
To speed the bug reporting process, please include basic environmental information with the bug post. For example, my post would start:
Intel iMac running OS X 10.5.7

20090617 Beta version of Universal FB Editor Available

-----Overview-----

The FBtoC team is pleased to announce beta testing for a new universal binary version of the FB Editor. The beta editor is a conversion of the older editor (included with the 'FB_5_2_Package') and thus retains most of the old editor's basic design and functionality. Many of the changes are internal to facilitate a universal binary build but some features were upgraded. Please note: Even though there are many proposed (but not implemented yet) upgrades (many internal) to this new UB editor, the team believes putting a Universal editor in the hands of the FB community is a higher priority than waiting for a 'finished' product. Even though it seems unlikely Apple will remove support of Rosetta in OS X 10.6, they have yet to clarify if it will be included, so having a universal binary editor protects the FB community for now.

-----Regarding Beta Status-----

Please note: this editor is a ***** BETA *****. This should NOT be interpreted as 'release'. There are identified unfixed bugs and new bugs/failures are possible. Proceed accordingly with saving your work, backups etc.

-----Changes-----

(1) Mac Intel users will generally notice a lot more zip because of the native executable. Speed increases for opening files (via double clicks on projects/files and via open dialogs) is particularly noticeable for me.

(2) The Find/Change dialog was improved and now works more reliably. Also, the Find/Change window is resizable.

(3) The split window (index on left) help window is used to display help. The content of help has been revamped to reflect FB5 instead of older FB, but is not complete and a work in progress. Other help manuals used to be available on the help window but are not in this version. Again, in progress.

(4) Several broken features in the old editor were fixed. For example, quadruple click on a long if now correctly highlights the entire structure from long if to end if. This also works for similar Start/End structures like while/wend, local fn/end fn etc.

-----Installation-----

- (1) Unzip the executable which is named 'FutureBasic_5_3beta' ---do NOT rename
- (2) Put it in the same folder(which is typically 'FB_5_2_Package' for the current release of FB5) as your current 'FutureBasic 5' application.
- (3) Launch 'FutureBasic_5_3beta' instead of 'FutureBasic 5'

Notes:

- (a) The beta editor requires FB5 and does not support FB4 or older FB releases
- (b) The beta editor is built with debugging symbols turned on to provide more debugging information in the event of a failure. The included debugging symbols increase the size of the executable but should not otherwise impact operation.

20090612 FutureBasic 5.2 and FBtoC 1.2

Editor is unchanged.

Bug fixes in FBtoC:

431	Closed	string stack bug from 'open "O", 1, name, @parFSRef'
432	Closed	'include "foo.c"' could give bad C code
433	Closed	log window shrinks each time FBtoC is launched
435	Closed	out-of-order menu items become spacers
436	Closed	hang from 'if 1 then pen 1 else pen 2'
437	Closed	Overlong path to project file should give helpful error
438	Closed	allow 16 parameters in function definitions
439	Closed	array bounds error translating 'for i=1to 6'
441	Closed	spurious translation error from 'stop fSpec.name'
442	Closed	stale gcc warning shown by editor

New feature:

p.0@ dereferences a pointer to a 64-bit integer (like p.0& for 32 bit)

New examples:

FB_5_2_Examples > Graphics > OpenGL > OpenGLTestBed
FB_5_2_Examples > Graphics > QuickDraw > (de)convolve img
FB_5_2_Examples > Miscellaneous > SortDescriptor

20090321 FutureBasic 5.1.5 and FBtoC 1.1.10

Editor changes:

- [1] An unsaved standalone file can now be built and run (Command-R); see note 1 below.
- [2] Compile Info window shows progress of both translation and compilation.

Bug fixes in FBtoC 1.1.10:

426	Closed	inkey\$ does not return
427	Closed	val&("01") and val&(" 1") should return 1
428	Closed	spurious error from 'if...then edit\$(1) = "" else stop'
429	Closed	'text' statement with no parameters hangs FBtoC
430	Closed	Tlbox glu.incl could not be included

New example:

FB_5_1_5_Examples > Text > HITextView+FontPanel > HITextView
An HITextView with the FontPanel allows multistyled text with shadow, colour, double-underline and other eye-candy.

Note 1:

This is a normal Run in every way, except that the various files required by FBtoC are in your TemporaryItems folder, which (in OS X 10.5) is somewhere like /private/var/folders/88/88AKX+F62RWmd++8ZOxBkU+++TI/TemporaryItems.

20090216 FutureBasic 5.1.4 and FBtoC 1.1.9

Editor 5.1.4 changes

1. Scripts '#if 1/#endif' and '#if 0/#endif' fixed.

FBtoC 1.1.9 changes

1. Typed functions without a suffix are allowed, for example
local fn Name as Str255
end fn = "Ann"
2. FB4 kludges removed from toolbox declarations (see below).
3. The setting 'Use precompiled header' works for universal builds.
4. The build system determines which pieces of a large program need to be compiled, and compiles only those pieces.

Bug fixes

415	Closed	Make Xcode Project wrongly quotes/escapes nib names
417	Closed	dropping file on FBtoC spoils log window position
420	Closed	Allow files with no OSType to be opened.
424	Closed	input # bug
425	Closed	record # should set file mark when extending

FB4 kludges removed

The Carbon C definition of these five functions includes a 64-bit parameter, which in FB4 had to be split into two 32-bit halves:

```
HIViewFindByID()  
HIViewSetID() // rarely used  
CreateMouseTrackingRegion()  
HIViewNewTrackingArea()  
HIViewChangeFeatures() // rarely used
```

You will have to change your code from, for example:

```
fn HIViewFindByID( fn HIViewGetRoot( w ), myHIViewID.signature, myHIViewID.id, @myView ) // old  
syntax
```

to the simpler correct form:

```
fn HIViewFindByID( fn HIViewGetRoot( w ), myHIViewID, @myView ) // FB5.1.4 syntax
```

20090109 FutureBasic 5.1.3 and FBtoC 1.1.8

Editor 5.1.3 is unchanged from 5.1.2.

FBtoC 1.1.8 changes:

'on FinderInfo fn Xxxx' implemented [see FB_5_1_3_Examples > Files > FinderInfo]
The Headers file Util_CE.incl has been extensively commented for easier use.

New Headers files and examples:

Util_EUTCKeyFilter.incl	[see FB_5_1_3_Examples > Text > EUTC Key filter]
Util_Toolbar.incl	[see FB_5_1_3_Examples > Carbon Toolbar > The easy way]
Tlxb_HITheme.incl	[see FB_5_1_3_Examples > Controls > HITheme_demos]
Tlxb_CoreText.incl	[see FB_5_1_3_Examples > Text > CoreText xxxx]

CoreGraphics headers changes

The special FB4 syntax for a handful of CG functions (CGRectMake, CGRectOffset, CGContextGetCTM ...) has been discarded. You will need to change the way you call these functions in your source code. For example:

```
fn CGRectMake( r, 0.0, 0.0, 10.0, 20.0 ) // old FB4 syntax  
r = fn CGRectMake( 0.0, 0.0, 10.0, 20.0 ) // new FB5 syntax similar to C
```

Additions to the CG headers:

```
CGDataProvider.incl  
CGError.incl  
CGFont.incl
```

```
CGGradient.incl
CGPath.incl
CGPDFArray.incl
CGPDFDictionary.incl
CGPDFDocument.incl
CGPDFObject.incl
CGPDFPage.incl
CGPDFStream.incl
CGPDFString.incl
```

The FB4 'local fn' wrappers have been removed from all CoreGraphics functions. This change makes 'fn' optional when calling CG procedures (functions that do not return a value).

```
fn CGContextFillRect( ctx, r ) // 'fn' formerly required; now optional
CGContextFillRect( ctx, r ) // new simpler syntax
```

Bug-fixes:

```
370 button statements uncheck a checked checkbox
404 scroll button mishandles Rect param at creation
406 key chars with high bit set are negative in _evKeys dialog event
409 I-beam cursor persists after click in FBtoC Log window
410 gcc error from struct/record field named 'index'
412 gcc error from pi if prefs 'Allow dim a#,a$,a%"
413 val&() should return 0 if first char is alphabetic
```

20081117 FutureBasic 5.1.2 and FBtoC 1.1.6

Important change in HandleEvents

The default behaviour of HandleEvents is now to block (i.e. not return) unless an event is dispatched.

It is no longer necessary (but is harmless), to suppress null events with:

```
poke long event - 8, 0xFFFFFFFF // no null events
```

The original behaviour of returning 30 times a second, if required for old polling code, can be restored by:

```
poke long event - 8, 2 // null events every 2 ticks, like FB4
do
fn PollForSomething
HandleEvents
until gFBQuit
```

Editor (FutureBasic 5.1.2) changes

Fixes to text display in the Errors window.

A bug has been fixed that caused the cursor to change to an arrow over the Scripts Palette even when the latter was hidden.

FBtoC 1.1.6 changes

1. Default behaviour of HandleEvents changed.
2. The rarely-used 'system' statement is no longer equivalent to 'end'. Please change your code to use 'end'.
3. The 'menu' statement now allows an optional 5th parameter (commandID).
4. New/revised Headers files: Tlhx CFBase.incl, Tlhx CFString.incl, Tlhx CFArray.incl, Tlhx CFData.incl, Tlhx CFDate.incl, Tlhx CFNumber.incl. These header files are more complete than previously; if you get redefined function errors for your own CFxxxx declarations, just comment out your declarations.
5. New Headers file: Util_Containers.incl [for use see FB_5_1_2_Examples > Containers > Util_Containers example].
6. New Headers file: Util_Toolbar.incl [for use see FB_5_1_2_Examples > Carbon Toolbar > Toolbar Example].
7. FB_5_1_2_Examples > Sorting [FB3/FB4 CD examples updated for FB5]
8. build_temp folders created by FBtoC are excluded from Time Machine backups.

Bug fixes

390 Closed left\$\$, mid\$\$, right\$\$ give _zTrue when comparing chars
391 Closed mid\$\$ (c, 1) gives "comma expected error"
392 Closed scrollbar of console app lacks down-arrow
393 Closed garbled error from proc "NoSuchFunction"
394 Closed prHndl..prInfo.rPage fields are byte-reversed on Intel
396 Closed default edit field is _framedNoCR instead of _framed
398 Closed can't access 2-D array inside record
399 Closed def apndstr has endian bug (on Intel)
401 Closed crash from print after front window closed
402 Closed 'Array bounds error' alert shows as 'Quit'

20081020 FutureBasic 5.1.1 and FBtoC 1.1.5

Editor (FutureBasic 5.1.1) changes

-
1. Script Palette replaces Script submenus.
 2. Project window has Headers button, which opens the Headers folder in the Finder.

FBtoC 1.1.5 changes

-
1. New File menu item: Make Xcode Project.
 2. New feature and setting: Warn of unused functions.
 3. Domain and folder type constants (_kUserDomain, _kApplicationSupportFolderType...) are now defined.
 4. New Headers file Util_CE.incl provides a simplified method to implement Carbon Events in your code; for demos see FB_5_1_1_Examples > CarbonEvents.

Bug fixes

314 Closed make strings in *.c runtime localizable
378 Closed elided error message from include library "NoSuchFile"
381 Closed editor ignores some translation errors
382 Closed aliased source file prevents translation of main
383 Closed editor's Stop button does not affect build by FBtoC
384 Closed I-beam cursor stays after click in Prefs 'More gcc options'
385 Closed unusable CreateMouseTrackingRegion() in Tlbx CarbonEvents.incl
386 Closed 'picH = picture' gives translation error
387 Closed _useWFont fails for some appearance buttons

20090924 FutureBasic 5.1 and FBtoC 1.1.4

Important change in FBtoC

Until now, the build process would automatically copy every nib (*.nib) from the source folder into the built app package. This greedy and surreptitious copy risked the inadvertent inclusion of files like "OldRubbish.nib" and "embarrassingly bad.nib". Now, nibs are ignored unless specified in an 'include resources' statement or in the editor project window (see below).

This means that every project or standalone file that uses a nib interface needs modification. The most straightforward fix is to add a line of code resembling this:
include resources "main.nib"
for each nib used by your project.

Editor (FutureBasic 5.1) changes

-
1. New menu items give better integration with FBtoC

20080912 FutureBasic 5 and FBtoC 1.1.3

The FutureBasic 5 editor is FBtoC-only as previously discussed.

After downloading the FB 5 package, please double-click FBtoC.app then click Open in the "downloaded from the Internet" warning dialog. Otherwise this dialog may interfere with your first Run from the editor.

20080722 FBtoC 1.1.2

1.1.2 build 140

Two helpful additions:

- [1] VAList - Variable number of function arguments support and examples added
- [2] FBShell - Basic program shell with more modern nib and carbon events

Bug Fixes:

- 362 Closed 140 Error if 1 then does not produce FBtoC error message
- 363 Closed 140 Error menu 127, 1, _disable does not work in FBtoC
- 364 Closed 140 Error Menu cmd keys don't auto repeat when created with FB menu statement
- 366 Closed 140 Error Record# should extend fork size without error

20080618 FBtoC 1.1.1

New Features:

(1) The new NavDialog function, an FBtoC-only option, is similar to the FB Files\$ keyword but provides substantially more functionality. NavDialog offers new functionality via Navigation Services but the details are hidden in the C runtime, so it is easy to call NavDialog (and several other NavDialogxxxxxx helper functions). Several resources are available for learning to use NavDialog(). They are:

- (a) The demos - see Examples/Files/NavDialog_Demos,
- (b) FBtoC help - see Language Enhancements
- (c) Browsing the NavDialogxxxxxx functions in FilesDollarFunction.c and some new FSRef / CFURL helpers in FileHandling.c (both in the build_goodies folder)

(2) FBtoC now uses NavDialog(_kNavDialogChooseFile + _kNavDialogSheet,...) internally.

(3) FBtoC Help for include statement revised

(4) Internal AppleScript arrangements simplified for debugging in Xcode

Bug Fixes:

- 246 Closed PM sheets beep in FBtoC-built app
- 359 Closed Appearance button btnNum, _activeBtn did nothing
- 360 Closed NavDialog sheets retract at once

Closed files\$(_FSRefSave,...) now returns parent folder ref always

- 361 Closed open "0", 1, s, @fsRef fails if fsRef is parent folder but file exists
- 362 Closed Architecture: Current Mac should disable OS min/max pop-ups

20080525 FB 4.4.3

The change most folks will appreciate is the ability to compile directly with either the FBtoC translator/gcc compiler or with the FB compiler. It is available now on the command menu.

Download FB 4.4.3 [here](#)

New Features for FutureBASIC 4.4.3 (aka: changes since 4.4.2)

4.4.3 (24 May 2008)

[1] Command menu has Run [FBtoC]

Note: if you're planning to use the new Run [FBtoC] command from within the FB Editor, please refer to the "Connecting to FBtoC" document for one-time setup steps.

[2] Obsolete 680x0 and FAT items removed from command menu

[3] New Headers files, matching FBtoC:

Subs FloatByteSwapping.incl

Tlbox AppleHelp.incl

Util_AppleHelp.incl

[4] Rntm Appearance.incl update fixes little-arrows-wrong-way-counting

[5] Subs Appearance Btns.Incl update supports viewsize for scroll bars.

[6] Subs Compiler.Incl update suppresses double-beep when adding too much text to an edit field.

[7] Subs Graphics.incl fixed for PPC compile.

[8] Definition of CFStringAppendPascalString corrected in Tlbox CFString.incl.

[9] FB Compiler v4.6 no longer emits "Warning: Your source code is ignoring the result of this toolbox function".

Special note for OS X 10.5 users

Before using FB 4.4.3 for the first time, locate the new FB Compiler.app in the Finder (the path is: FB_pkg_4_4_3/FB Extensions/Compiler/FB Compiler.app) and launch it by double-clicking its icon. If an alert appears ("FB Compiler.app is an application which was downloaded from the Internet. Are you sure you want to open it?"), click the Open button.

P.S. Currently, the 4.4.3 version is only available in English.

20080524 FBtoC 1.1

New features:

[1] files\$ supports _FSRefFolder and _CFURLRefFolder as alternatives to _FSSpecFolder.
See Examples > Files > open/files\$ FSSpec/FSRef/CFURLRef

[2] MakeIntoFBWindow() has been improved to handle tracking controls correctly (scroll bars, little-arrows and sliders, both live and non-live).

The improvements required the code to be moved to the C runtime instead of being user FB code.

See Examples > Nib to FB wndNum & btnNum > Nib demo.bas

See also NibToFB.incl in the same folder, in which MakeIntoFBWindow() is now declared:

```
toolbox MakeIntoFBWindow( WindowRef w, SInt32 wNum )
```

The syntax fn MakeIntoFBWindow(wRef, wNum) is unchanged.

[3] FBtoC can now be invoked from the FutureBASIC editor (v4.4.3 whose release is expected soon).

New Header files:

Tlbg AppleHelp.incl and Util_AppleHelp.incl

Bug fixes:

355	Closed	files\$(_FSSpecFolder,...) should produce no-name variant FSSpec
356	Closed	nib sliders and little-arrows don't send dialog events
357	Closed	edit\$(n) = "ABC" blows string stack
358	Closed	'input #1, longvar' gives 0 for -ve number strings

This procedure converts a nib-based window and its controls to behave as though created programmatically with 'appearance window' and 'appearance button' statements. After a call to MakeIntoFBWindow(), the window has an FB wNum and its controls have FB btnNums, and generate FB dialog events.

20080508 FBtoC 1.0.9

The 1.0.9 release of FBtoC (English only) repairs some important bugs.

The French localization, already outdated in 1.0.8, became severely dysfunctional in 1.0.9 and has been temporarily removed.

Bug fixes:

351	Closed	'input #fileID, stringVar, singleVar, doubleVar' gives null string and 0s
352	Closed	'on mouse fn DoMouse' interferes with button handling
353	Closed	little-arrows and sliders should give correct FB dialog events
354	Closed	'edit field' blows string stack

20080424 FBtoC 1.0.8

New special constant _forLoopsAlwaysExecutedAtLeastOnce:

```
for j = 1 to 0
  print "Never get here"
next
override _forLoopsAlwaysExecutedAtLeastOnce = _true
for j = 1 to 0
  print "Do get here (legacy FB behaviour)"
next
```

Division symbols:

/ and double-backslash: always mean the same thing (integer division if appropriate)

\ always means force floating point division

Bug fixes:

345	Closed	close box doesn't close window
347	Closed	performance issue with container += string + string
348	Closed	crash from edit\$(1, _maxInt, _maxInt) = "a"
349	Closed	performance issue with hex\$()

20080327 FBtoC 1.0.7

New feature:

- for/next loops may be executed zero times
for j = 1 to 0
 print "Never get here"
next

New examples:

Examples > Calculations > Counting for 1 second
Examples > CoreGraphics (Quartz) > CGBitmap demo

New documentation:

Build system caching.rtfcd [a file in the FBtoC_1.0.7 folder]

Bug fixes:

284	Closed	can't assign a string const to another
327	Closed	'print myUInt64Var/1024/1024/1024' prints 0
336	Closed	line continuation bug
337	Closed	def open "TYPE" does nothing
338	Closed	default file type should be 'TEXT'
339	Closed	crash from 'print test.r.left'
342	Closed	some labels become illegal C code
344	Closed	wrong number of decimal places sometimes printed

Open bugs:

314	Open	make strings in *.c runtime localizable
323	Open	input "prompt"; x does not work in console mode
343	Open	gcc error from pointer.offsetConst\$ arg to toolbox Str255 param

20080307 FBtoC 1.0.6

New feature:

- menu item "Move build_temp to Trash"

New examples:

Examples > Mandelbrot set
Examples > Carbon Toolbar > Toolbar.proj

Bug fixes:

077	Closed	fix the 'as pointer/Handle to type' mess
212	Closed	crash when global var 'err' is used (OS X 10.4)
216	Closed	app gets generic icon if project name contains '&'
329	Closed	'stop' decrements the string stack
330	Closed	'picture' statement can crash in OS X 10.2.8
331	Closed	gcc error from global vars j0, j1, jn, y0, y1, yn
332	Closed	Template error if FB-built app is in source directory
333	Closed	spurious error from 'long color' inside 'long if' block
334	Closed	error: 'index' redeclared as different kind of symbol

Notes:

With bug #077 closed, you can now write code like this:

```
'-----  
#define MyFunkyType as Handle to Rect  
  
begin record DemoRec  
dim as MyFunkyType h1, h2  
dim as pointer to Rect p1, p2  
end record
```



```
dim as DemoRec myFunkyVar
```

```
myFunkyVar.h1 = fn NewHandle( sizeof( Rect ) )
myFunkyVar.h1..top = 99
'-----
```

which was impossible in FB owing to bugs (that remained unfixed throughout the compiler's entire lifespan).

FBtoC also fixes this related FB bug:

```
'-----
dim myPtrToSpec as .FSSpec
print "Should be 4 because it's pointer:", sizeof( myPtrToSpec )
'-----
```

20080222 FBtoC 1.0.5

New features:

- French localization of FBtoC
- simplified debugging with Xcode, via menu item File > Debug Last App

New example:

Examples > Debug > DemoCrash

DemoCrash crashes at the line:
poke address, value.

When run under Xcode's debugger, the crashing code and variable values are shown

20080205 FBtoC 1.0.4

New features:

on edit fn MyDoEdit
tekey\$ function
support for localization of string literals

New example:

Localizable strings

Bugs fixed:

304 Closed	int(1/3*100) evaluates as 0
306 Closed	print #fileID spc(10) prints nothing to file
307 Closed	comma-tabbing and tab() broken for print #fileID
308 Closed	remember Finder-dropped file for 'Open Last File' command
310 Closed	pos(fileID) returns nonsense
311 Closed	closing a _scrollVert button makes window(_width) off by 1
313 Closed	close box should not close window if 'on dialog fn
DoDialog' is in effect	
318 Closed	text statement can't be followed by a colon
319 Closed	gcc warning from picture statements
320 Closed	inc() / dec() can't be applied to record fields

20080129 FBtoC 1.0.3

Updated Headers file:

CFIndex.incl

Bugs closed:

279 Closed _scrollVert and _scrollHoriz scroll buttons not fully working
301 Closed custom-sized string vars can wrongly be Str255
302 Closed button value can't be set with 'appearance button...'
303 Closed files\$(_FSSpecFolder...) returns invalid FSSpec

20080126 FBtoC 1.0.2

Updated Headers file:
CFIndex.incl

New feature:

_scrollVert and _scrollHorz scroll buttons can now be created. They resize automatically with the window. But the apparent window size (from window(_width) etc) is not yet adjusted for presence of scroll bars.

Bugs closed:

283 Closed problems from source file with long name
293 Closed allow drag and drop from Finder
296 Closed 'input' gets character on key-up
297 Closed DIM 4 MTBold% should be a translation error
298 Closed _wndRefresh, _wndActivate FB events not received
299 Closed 'unterminated comment' gcc error for some division expressions
300 Closed FB dialog handler signals _evKey instead of _evCmdKey

20080116 FBtoC 1.0.1

Now implemented:

page lprint, read dynamic, write dynamic, scroll

New section in FBtoC Help:

Byte order

Headers files now available:

Subs AppleScript.incl, Tlbx CFByteOrder.incl

Bugs closed:

285 Closed 'window 1' resets print position to top left
287 Closed 'on mouse' disabled
289 Closed printed text wrongly split over several lines
292 Closed built app has old creation date
294 Closed last member of STR# resource not retrieved
295 Closed crash from input "test"; x

20080102 FBtoC 1.0

Changes since preview 1a126:

Translates 'on timer fn DoTimer'

SInt64 and UInt64 are now integer types, not records

Bugs closed:

265 'dim dynamic' fails for custom-sized string array
281 custom-sized string vars can't be suffixed with '\$'
282 allow kEventRawKeyRepeat to reach fn DoDialog as _evKey