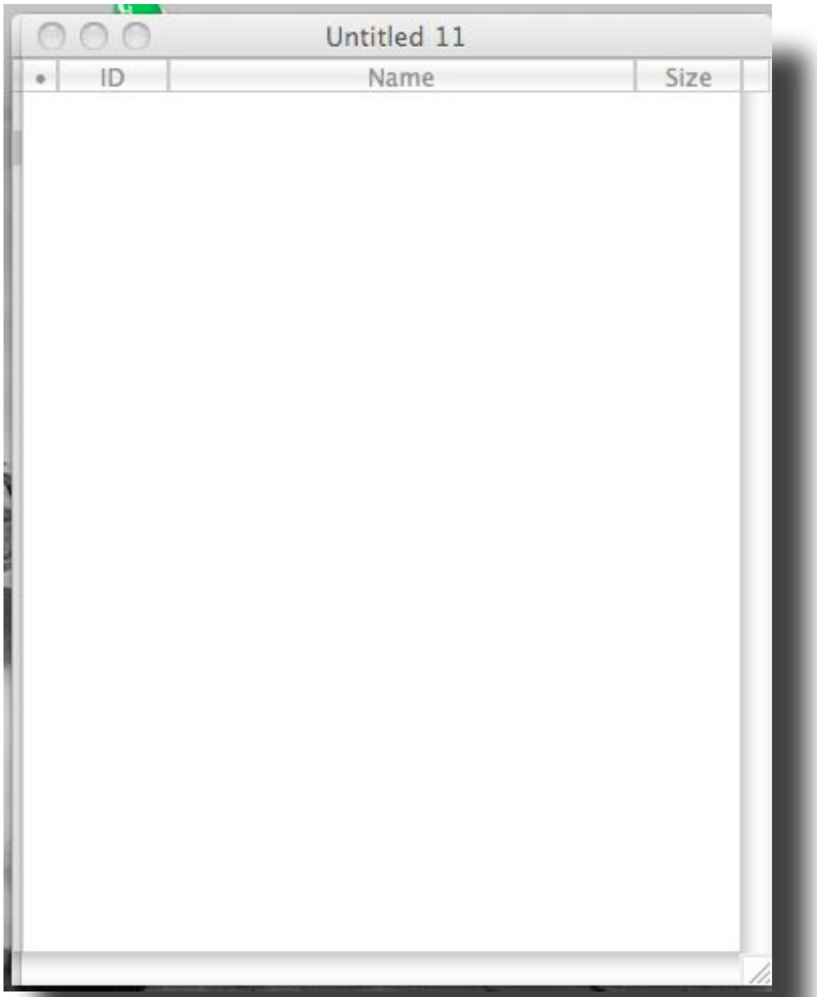
Disclaimer:
Use at your own risk. This worked for me, and I have been happy with the results. You may not.

Using ResFOOL, Create a new resource.

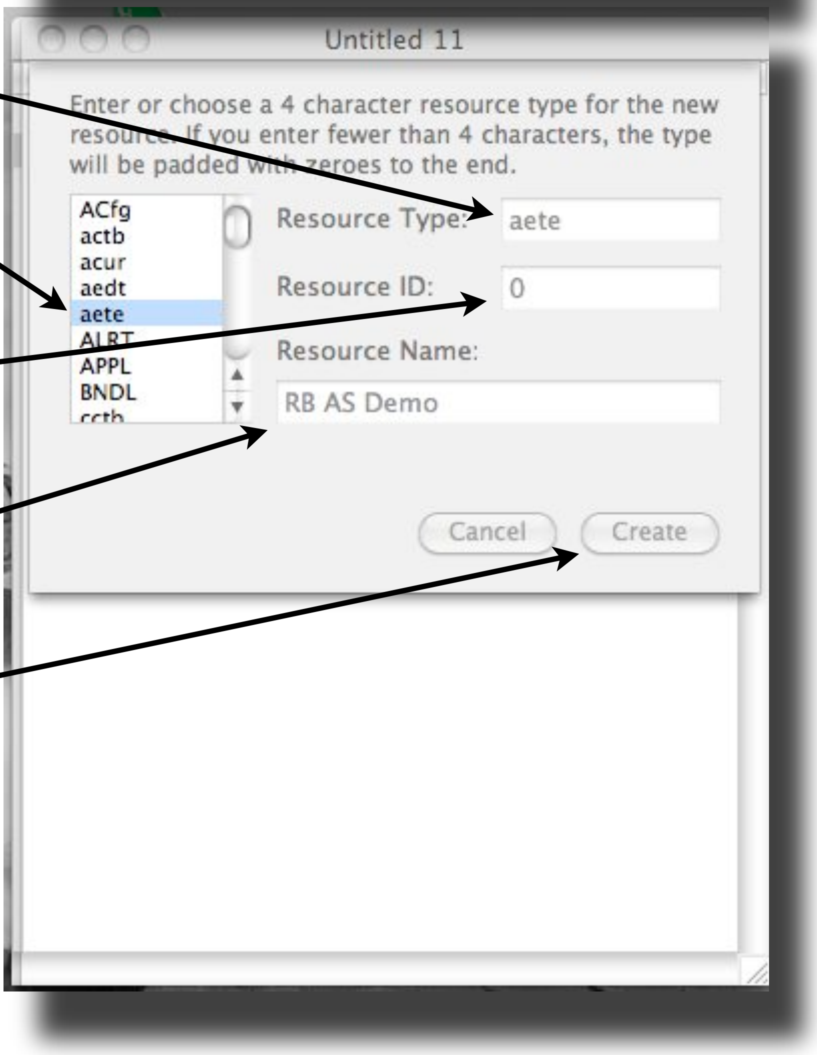


Select aete.

Change resource ID to 0

Give the dictionary (resource) a name.

Click on create



Enter or choose a 4 character resource type for the new resource. If you enter fewer than 4 characters, the type will be padded with zeroes to the end.

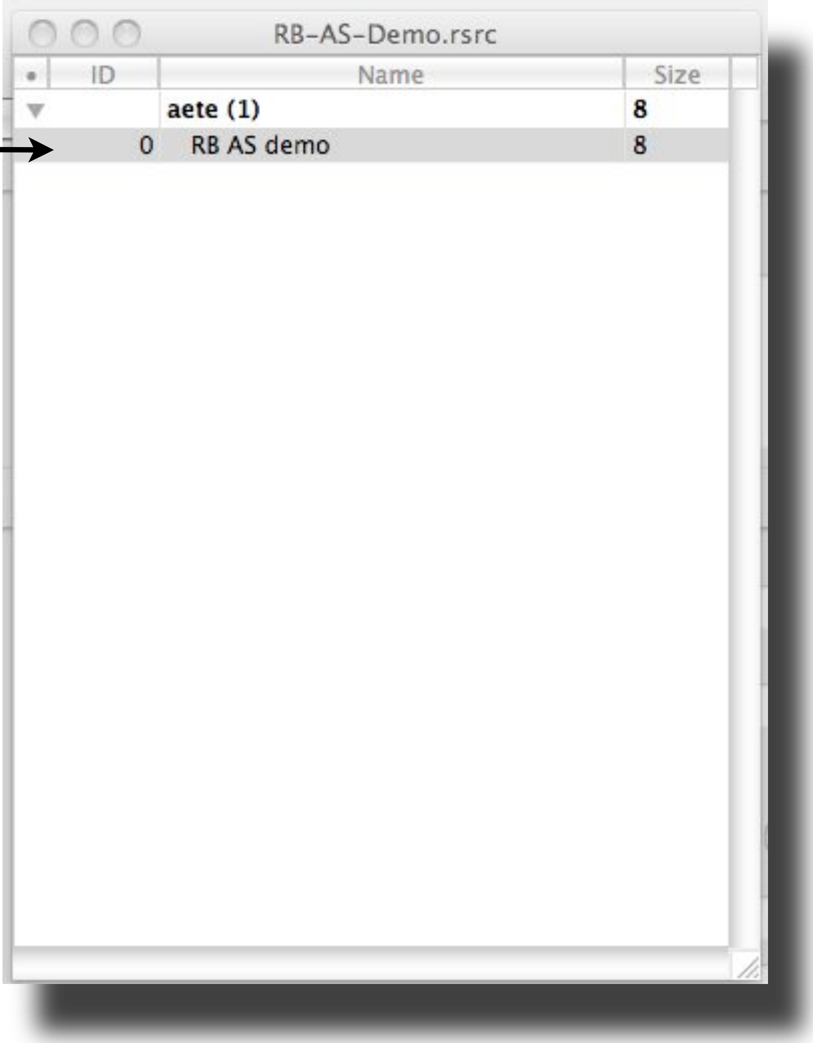
Resource Type: aete

Resource ID: 0

Resource Name:

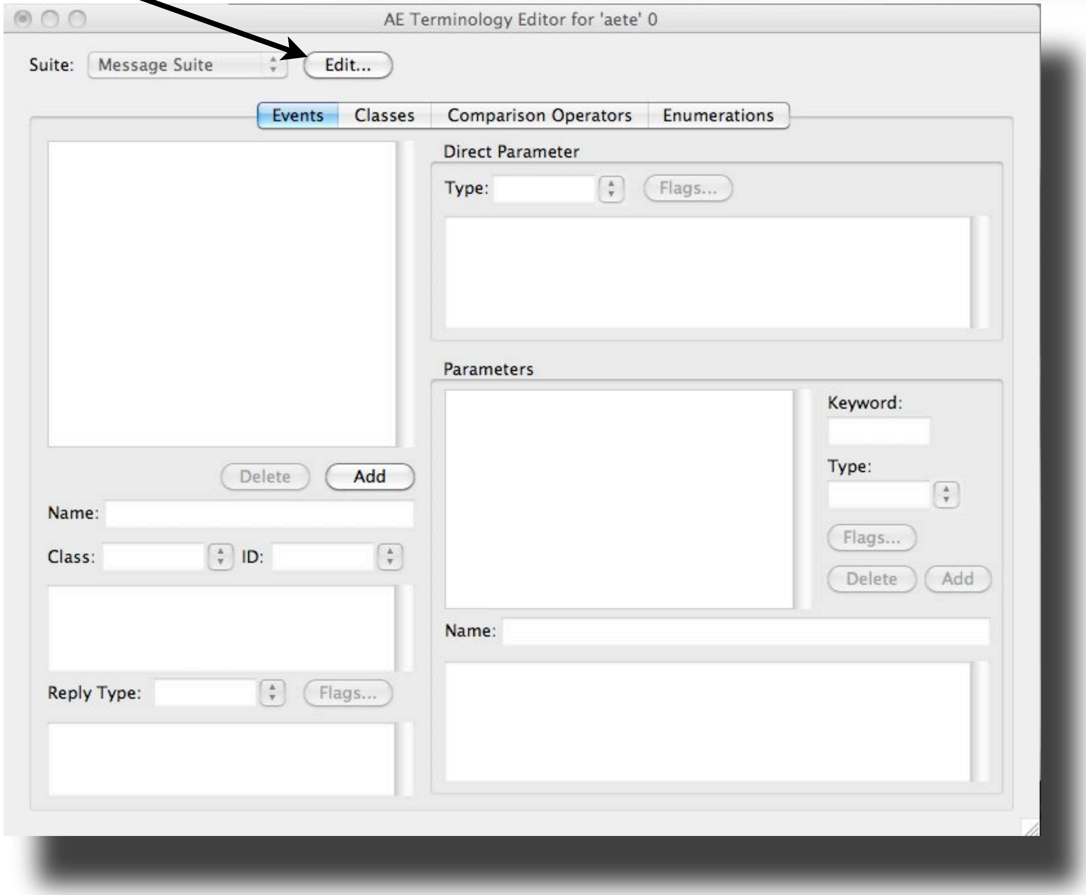
RB AS Demo

Double click the newly created resource (dictionary)



•	ID	Name	Size
▼		**aete (1)**	8
	0	RB AS demo	8

Click the Edit



Suite: Message Suite

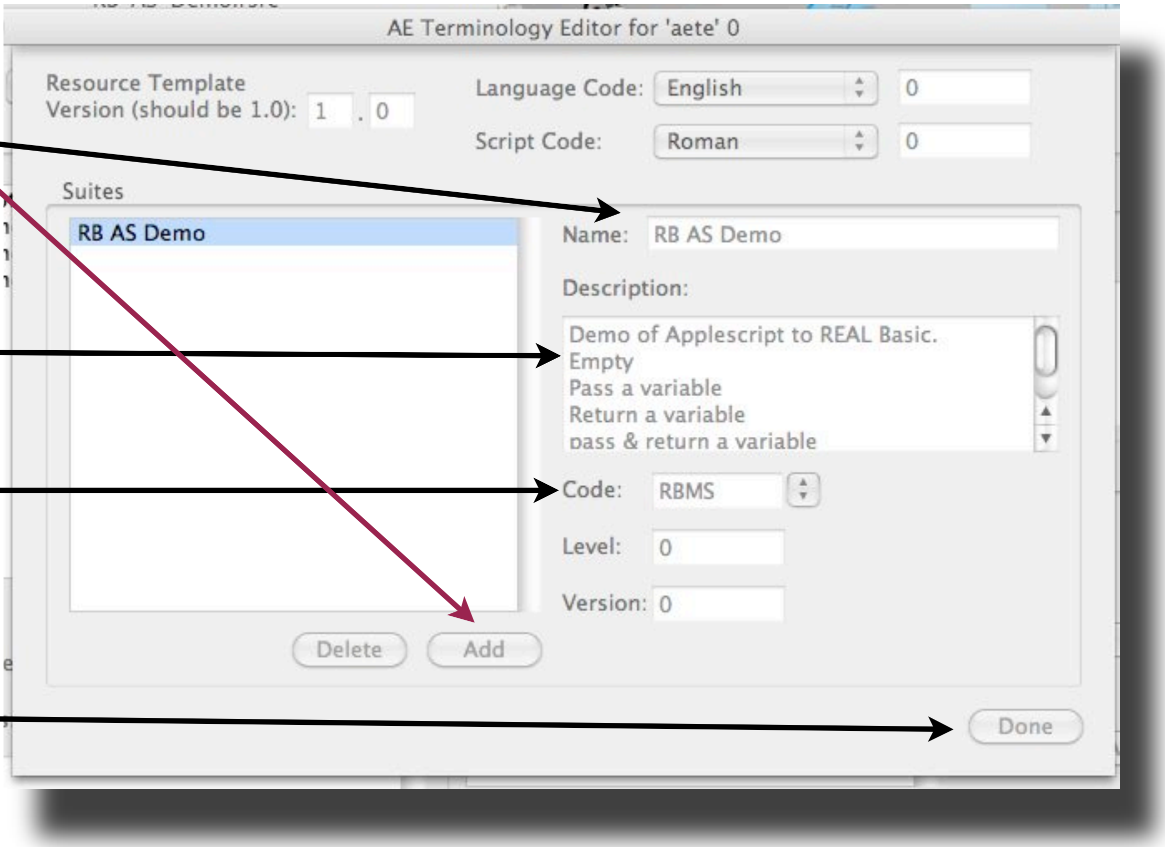
Click Add

Give the resource (dictionary) a name.

Dictionary description

Also give it a Code. (this code is the name of the resource (dictionary inside REAL Basic)

Click Done.



Resource Template Version (should be 1.0): 1 . 0

Language Code: English 0

Script Code: Roman 0

RB AS Demo

Name: RB AS Demo

Description:
Demo of Applescript to REAL Basic.
Empty
Pass a variable
Return a variable
pass & return a variable

Code: RBMS

Level: 0

Version: 0

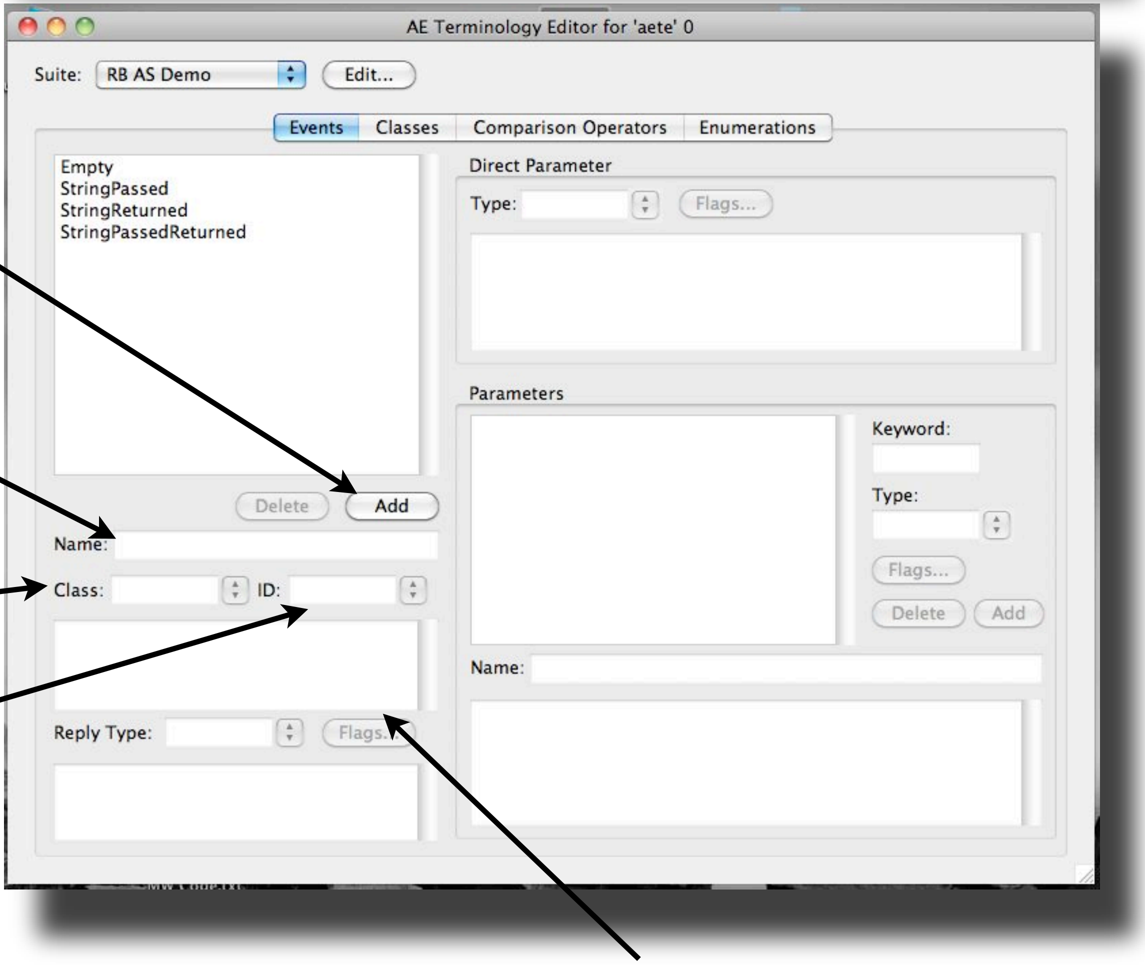
Here's the meat and potato part.

Click add to create a new function (handler, method take your pic)

The Name you give it will be the name used in AppleScript.

Class is the Code from above

The ID is the name REAL Basic will use/ see when its called. (4 Characters)



Suite: RB AS Demo

Empty
StringPassed
StringReturned
StringPassedReturned

Description displayed when dictionary is opened in Apple Script Editor

Passing and returning

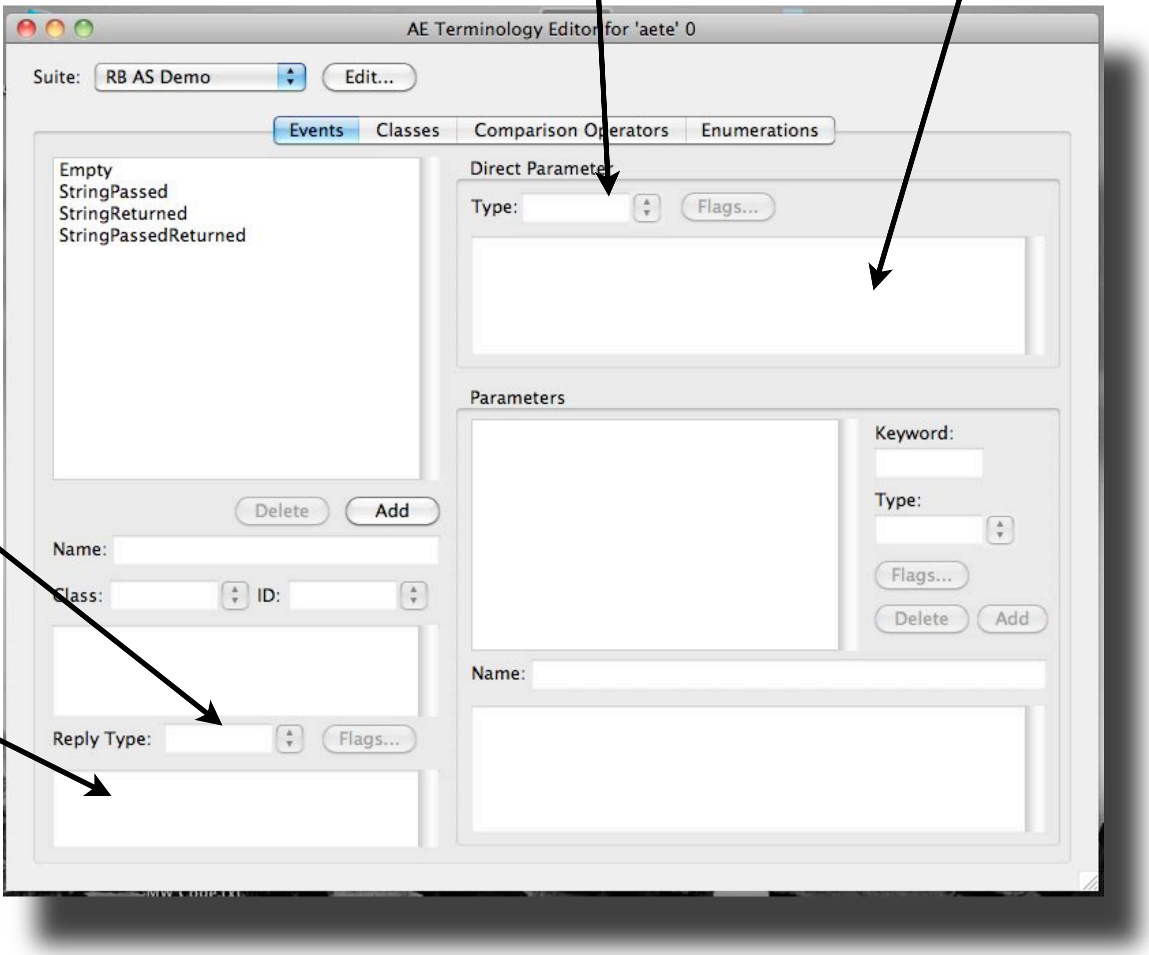
I'm only using strings since thats all AppleScript deals with. Changing to integers or numbers I do in the program(s).

To pass a string into REAL Basic, Type in TEXT.

Description of passed String

Over in REAL Basic, here's the code to get things rolling. Its in the APP > EventHandler > HandleAppleEvent
You see the ID from above is the hook to RB. And the RBMS is the the ball of wax.

Apple Event that kicks off



Suite: RB AS Demo

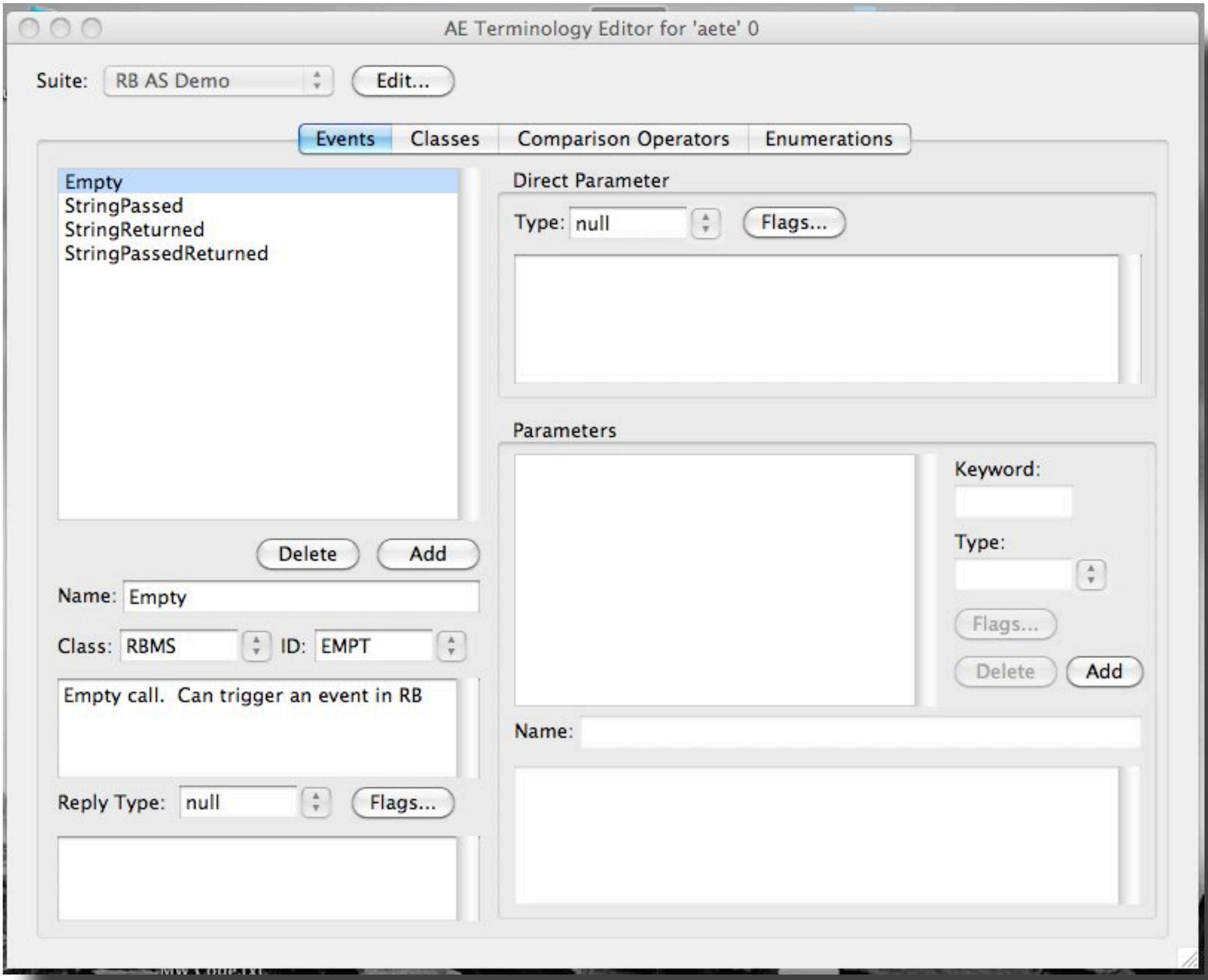
Empty
StringPassed
StringReturned
StringPassedReturned

To have a string returned, change reply type to TEXT. (TEXT wasn't in the list, so I just typed it in and its working here)

Description of returned

Now for Examples

Empty function
Can be used to trigger something in RB



Suite: RB AS Demo

Empty
StringPassed
StringReturned
StringPassedReturned

Name: Empty

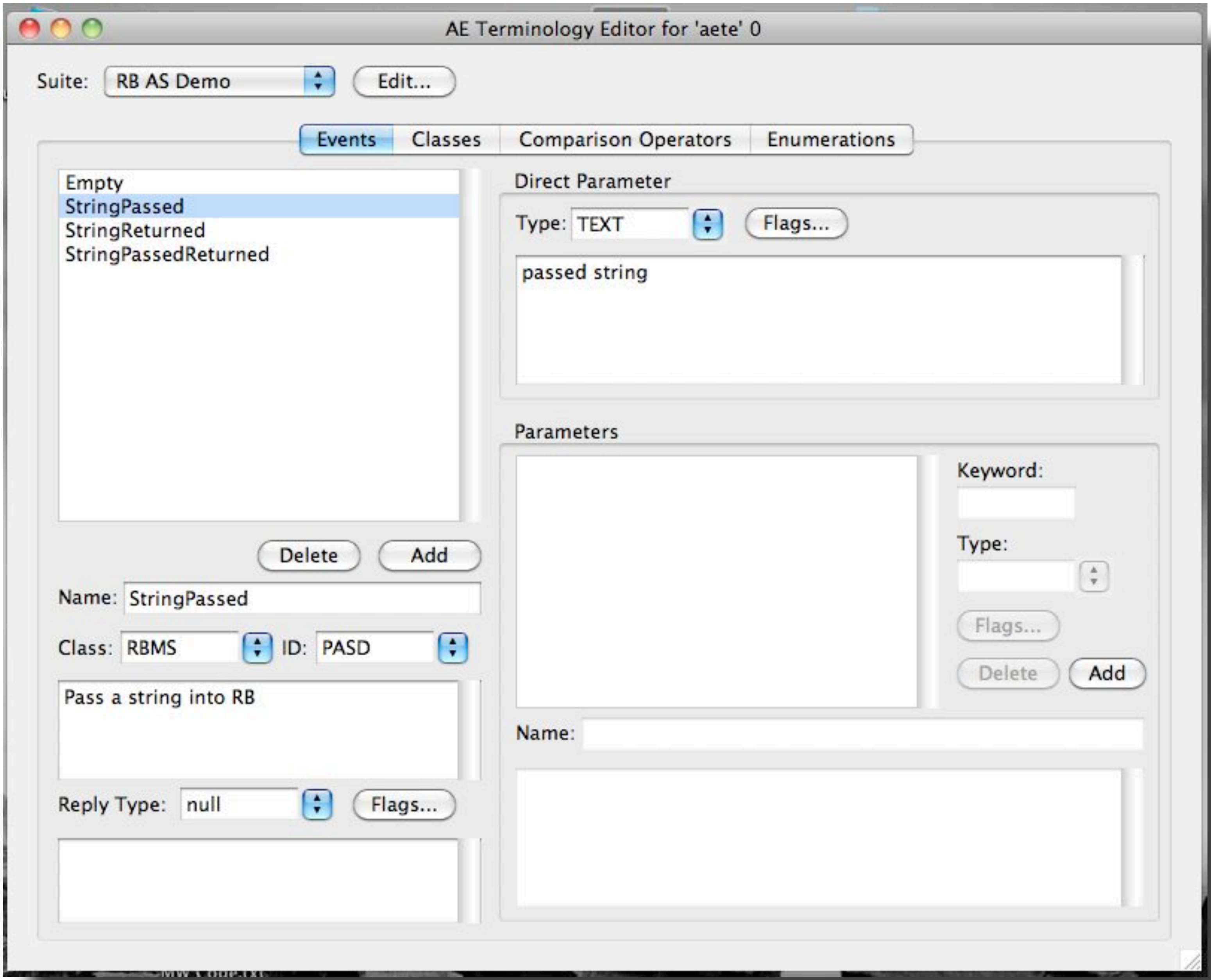
Class: RBMS ID: EMPT

Empty call. Can trigger an event in RB

Reply Type: null

Type: null

Passing a String to RB



AE Terminology Editor for 'aete' 0

Suite: RB AS Demo

Empty
StringPassed
StringReturned
StringPassedReturned

Name: StringPassed

Class: RBMS ID: PASD

Pass a string into RB

Reply Type: null

Direct Parameter

Type: TEXT

passed string

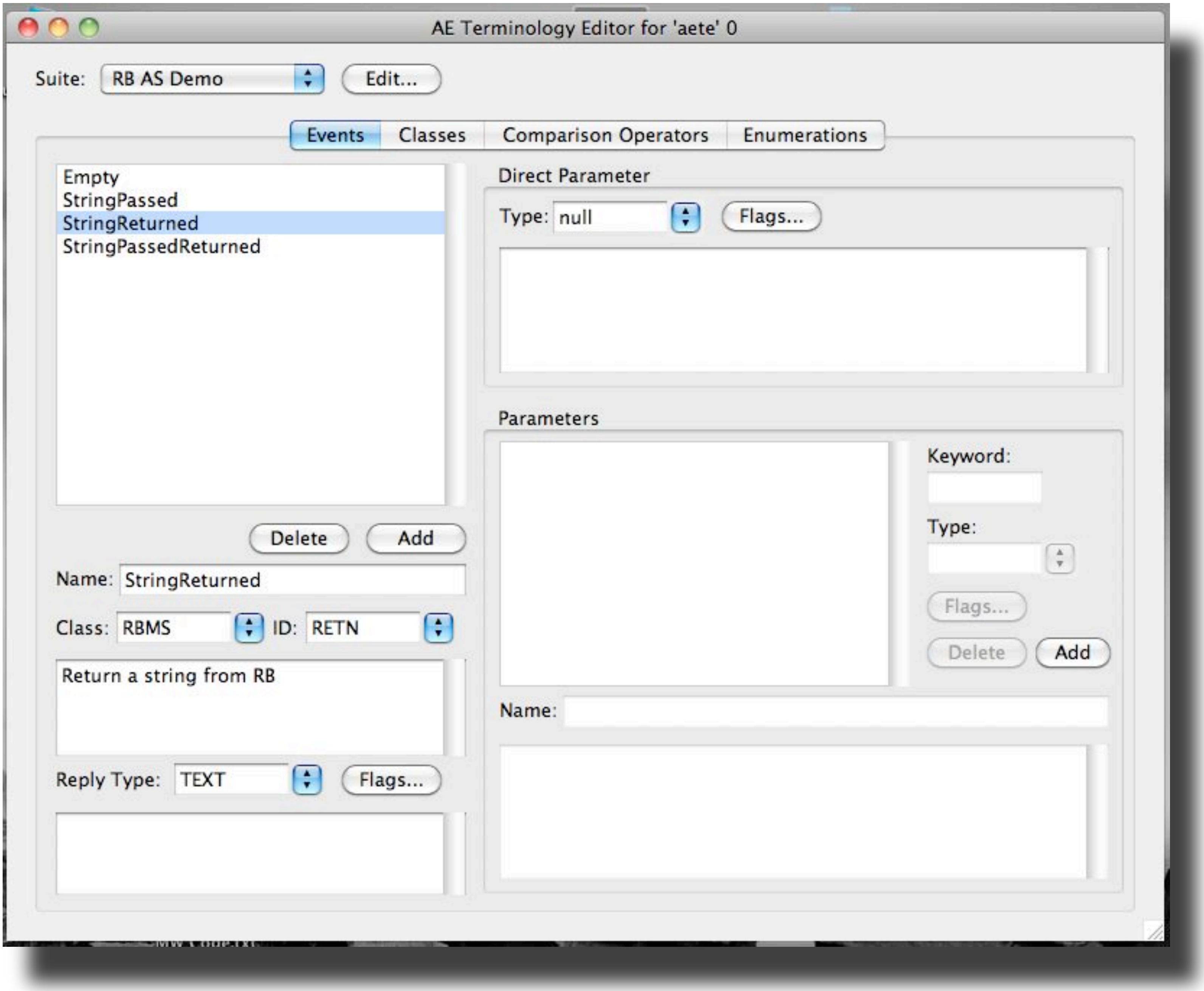
Parameters

Keyword:

Type:

Name:

Return a String to AppleScript



Suite: RB AS Demo

Name: StringReturned

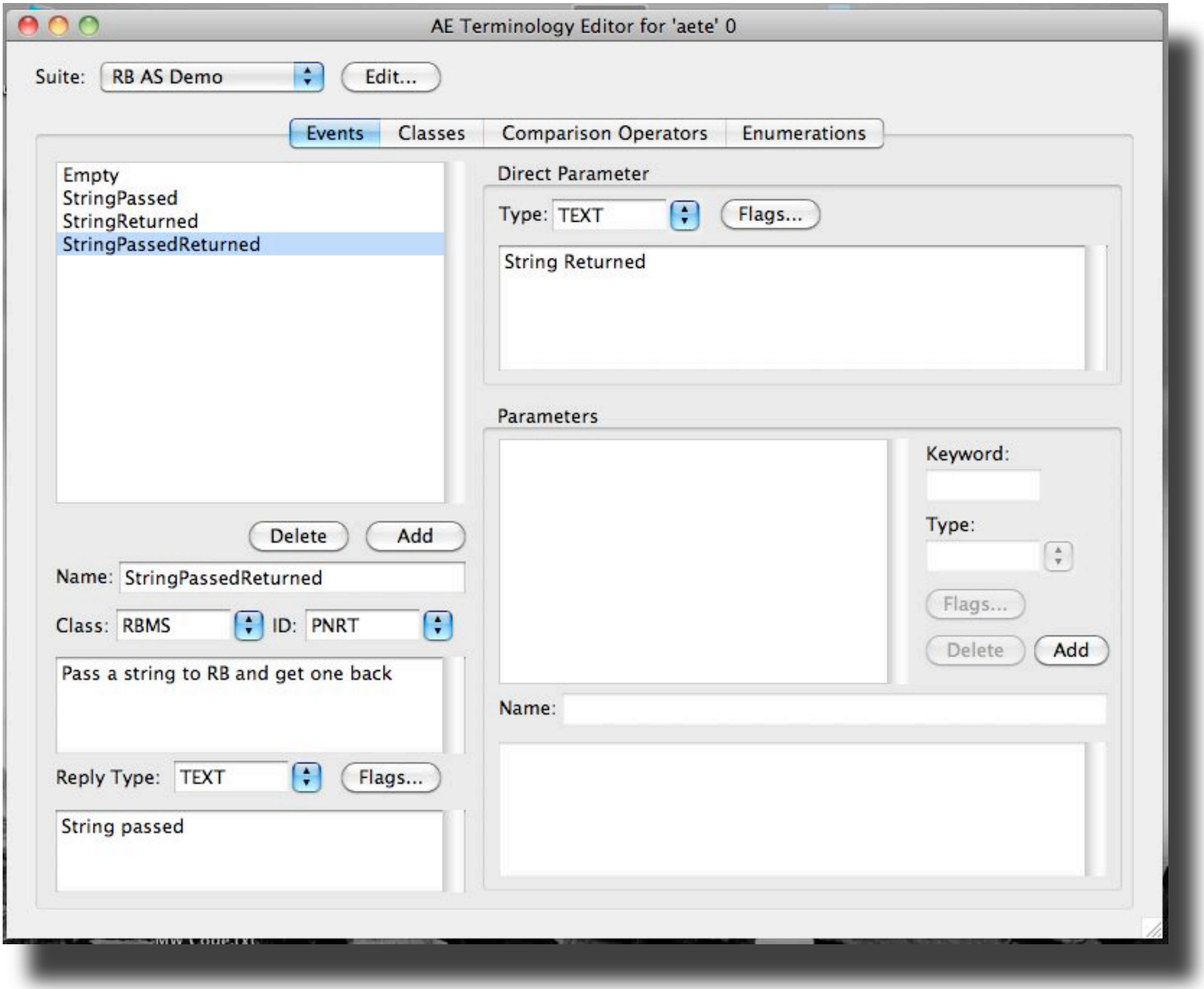
Class: RBMS ID: RETN

Return a string from RB

Reply Type: TEXT

Type: null

Pass a String to RB and return one to AppleScript


AE Terminology Editor for 'aete' 0

Suite: RB AS Demo

Events:
- Empty
- StringPassed
- StringReturned
- StringPassedReturned

Name: StringPassedReturned

Class: RBMS ID: PNRT

Pass a string to RB and get one back

Reply Type: TEXT

String passed

Direct Parameter

Type: TEXT

String Returned

Parameters

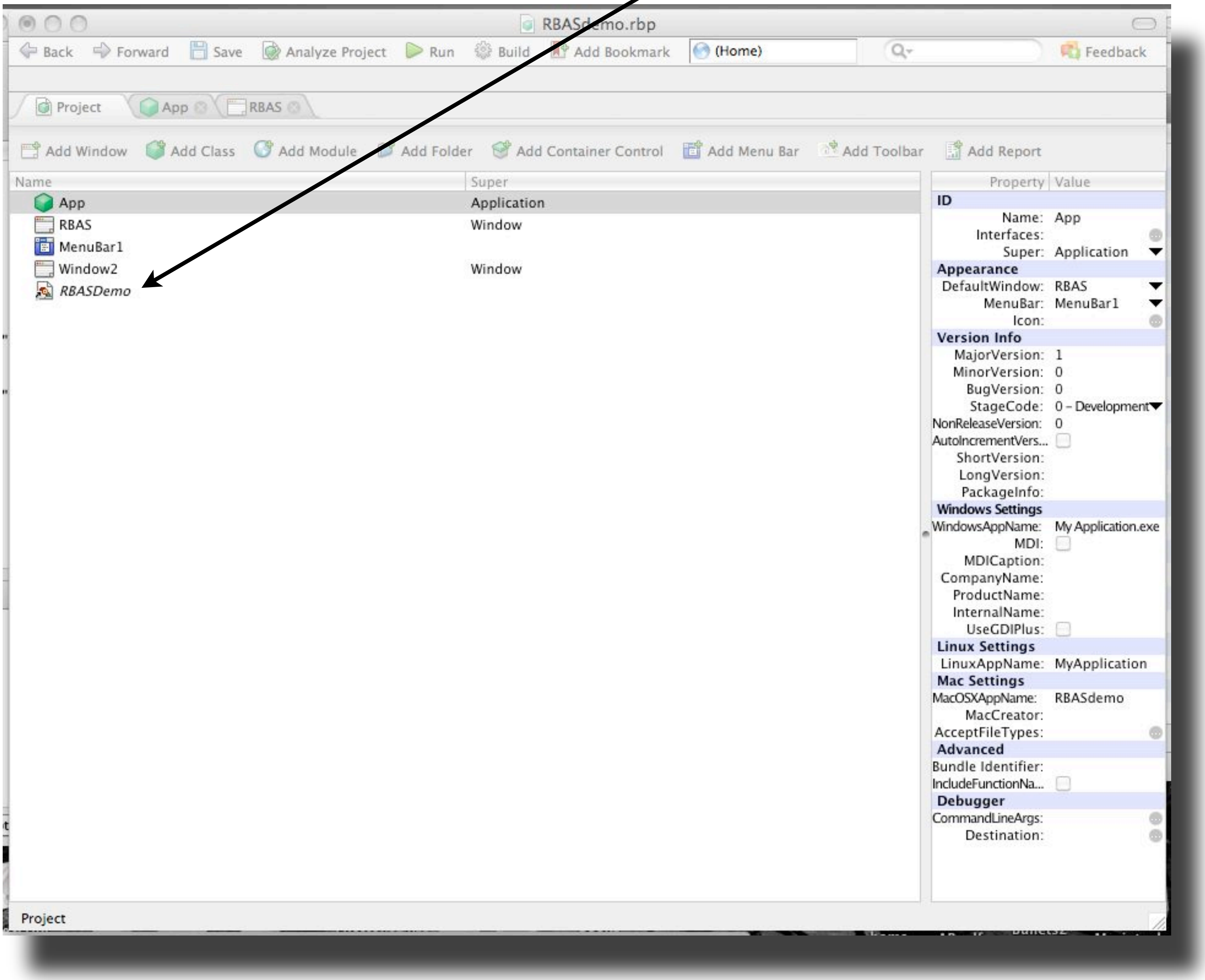
Keyword:

Type:

Name:

REAL Basic section:

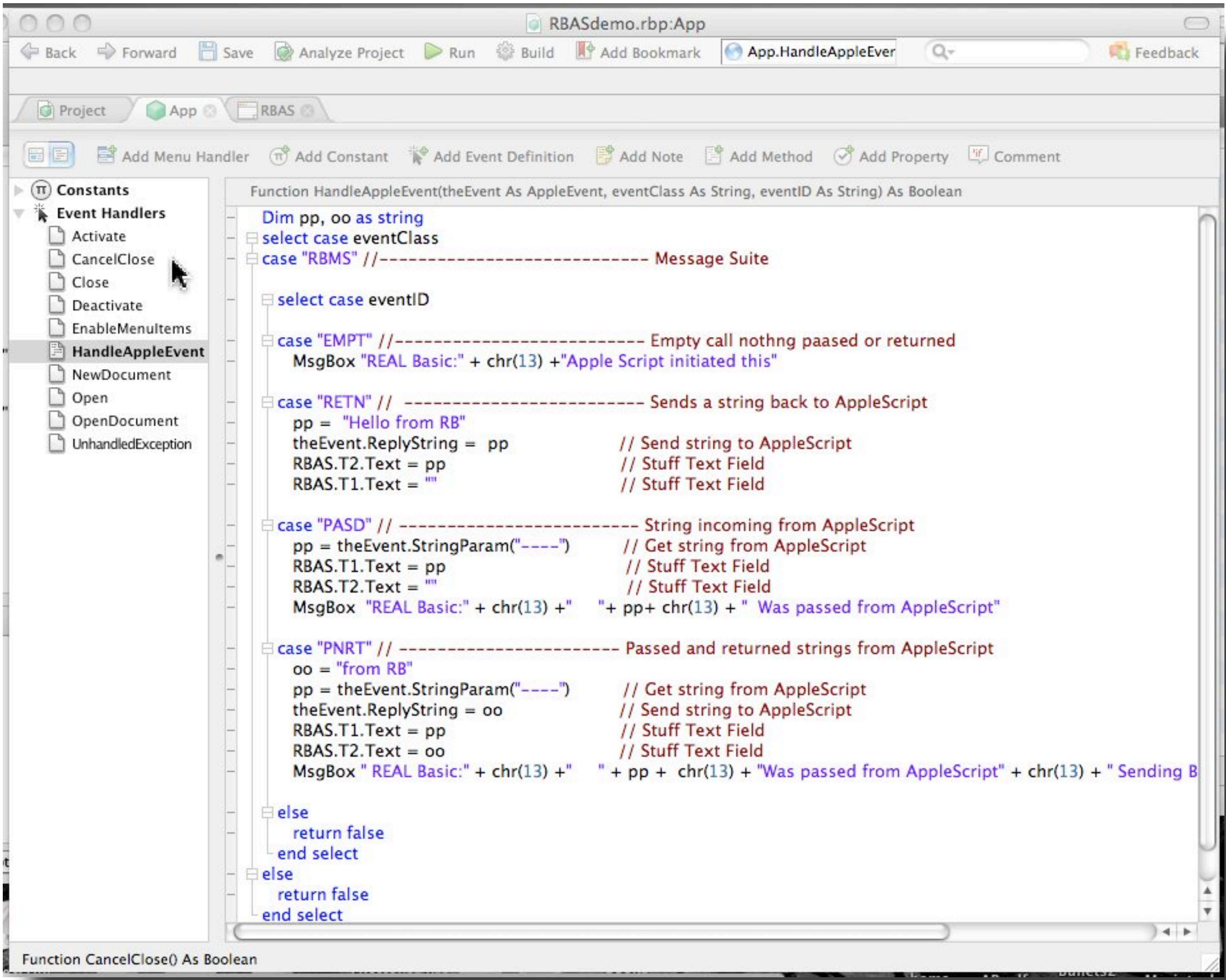
Save Resource file and add to Project



Name	Super
App	Application
RBAS	Window
MenuBar1	
Window2	Window
RBASDemo	

Property	Value
ID	
Name:	App
Interfaces:	
Super:	Application
Appearance	
DefaultWindow:	RBAS
MenuBar:	MenuBar1
Icon:	
Version Info	
MajorVersion:	1
MinorVersion:	0
BugVersion:	0
StageCode:	0 – Development
NonReleaseVersion:	0
AutoIncrementVers...	☐
ShortVersion:	
LongVersion:	
PackageInfo:	
Windows Settings	
WindowsAppName:	My Application.exe
MDI:	☐
MDICaption:	
CompanyName:	
ProductName:	
InternalName:	
UseGDIPlus:	☐
Linux Settings	
LinuxAppName:	MyApplication
Mac Settings	
MacOSXAppName:	RBASdemo
MacCreator:	
AcceptFileTypes:	
Advanced	
Bundle Identifier:	
IncludeFunctionNa...	☐
Debugger	
CommandLineArgs:	
Destination:	

Add the REAL Basic Code



```
Function HandleAppleEvent(theEvent As AppleEvent, eventClass As String, eventID As String) As Boolean
Dim pp, oo as string
select case eventClass
case "RBMS" //---------------------------- Message Suite

  select case eventID

  case "EMPT" //-------------------------- Empty call nothng paased or returned
    MsgBox "REAL Basic:" + chr(13) +"Apple Script initiated this"

  case "RETN" //  ------------------------ Sends a string back to AppleScript
    pp =  "Hello from RB"
    theEvent.ReplyString =  pp                // Send string to AppleScript
    RBAS.T2.Text = pp                         // Stuff Text Field
    RBAS.T1.Text = ""                         // Stuff Text Field

  case "PASD" // ------------------------- String incoming from AppleScript
    pp = theEvent.StringParam("----")          // Get string from AppleScript
    RBAS.T1.Text = pp                          // Stuff Text Field
    RBAS.T2.Text = ""                          // Stuff Text Field
    MsgBox  "REAL Basic:" + chr(13) +"     "+ pp+ chr(13) + "  Was passed from AppleScript"

  case "PNRT" // ---------------------- Passed and returned strings from AppleScript
    oo = "from RB"
    pp = theEvent.StringParam("----")          // Get string from AppleScript
    theEvent.ReplyString = oo                 // Send string to AppleScript
    RBAS.T1.Text = pp                         // Stuff Text Field
    RBAS.T2.Text = oo                         // Stuff Text Field
    MsgBox " REAL Basic:" + chr(13) +"     " + pp +  chr(13) + "Was passed from AppleScript" + chr(13) + " Sending B

  else
    return false
  end select
else
  return false
end select
```

AppleScript section:

(I didn’t give this RB project a name for for the AppleScript part, we will assume I named it RBASdemo, we will also assume that the ‘extras’ are named StringPassed, StringReturned and StringPassedReturned, with the ID’s of PASD, RETN, and PNRT respectively.)

(The line breaks in the display dialog lines have a \n to give a second line in the dialog message box, add these to the script)

```
-- RB demo

tell application "RBASdemo.debug"
	Empty
	delay 3
	beep
	StringPassed ("dude")
	delay 3
	beep
	set bb to StringReturned
	display dialog "AppleScript
" & "This was returned from REAL Basic: " & bb
	delay 3
	beep
	set bb to StringPassedReturned ("hello dude!")
	display dialog "AppleScript:
This was returned from REAL Basic:
" & bb


end tell
```

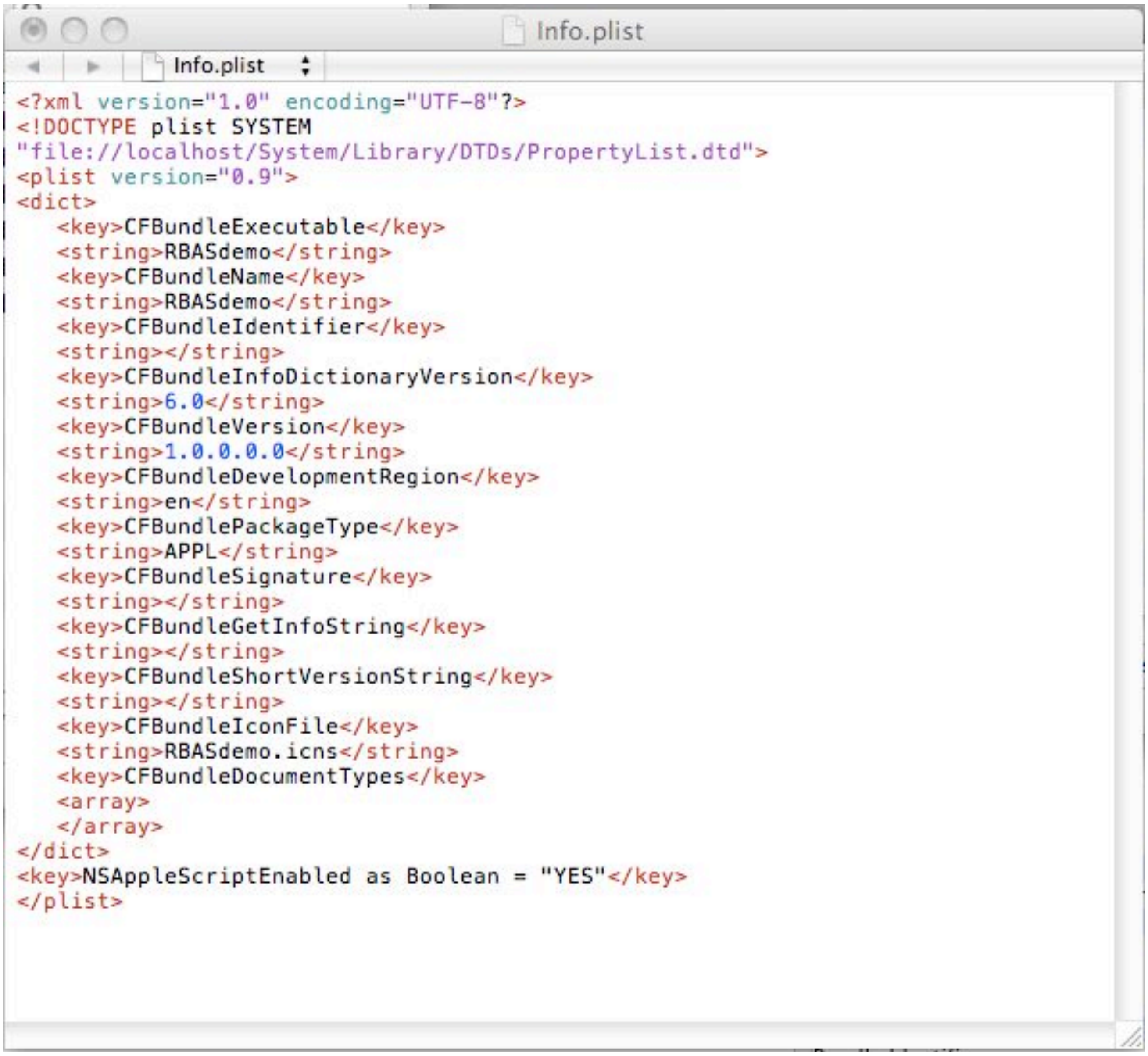
Notes:

TheAppleScript example is using the .debug, not a built application. Remove this for a built application.
If during your debugging, you encounter Applescript error about not finding the app or the like (like when it asks to search for the application) Just quit the AppleScript editor completely, run the app then restart Applescript editor and load file/script.

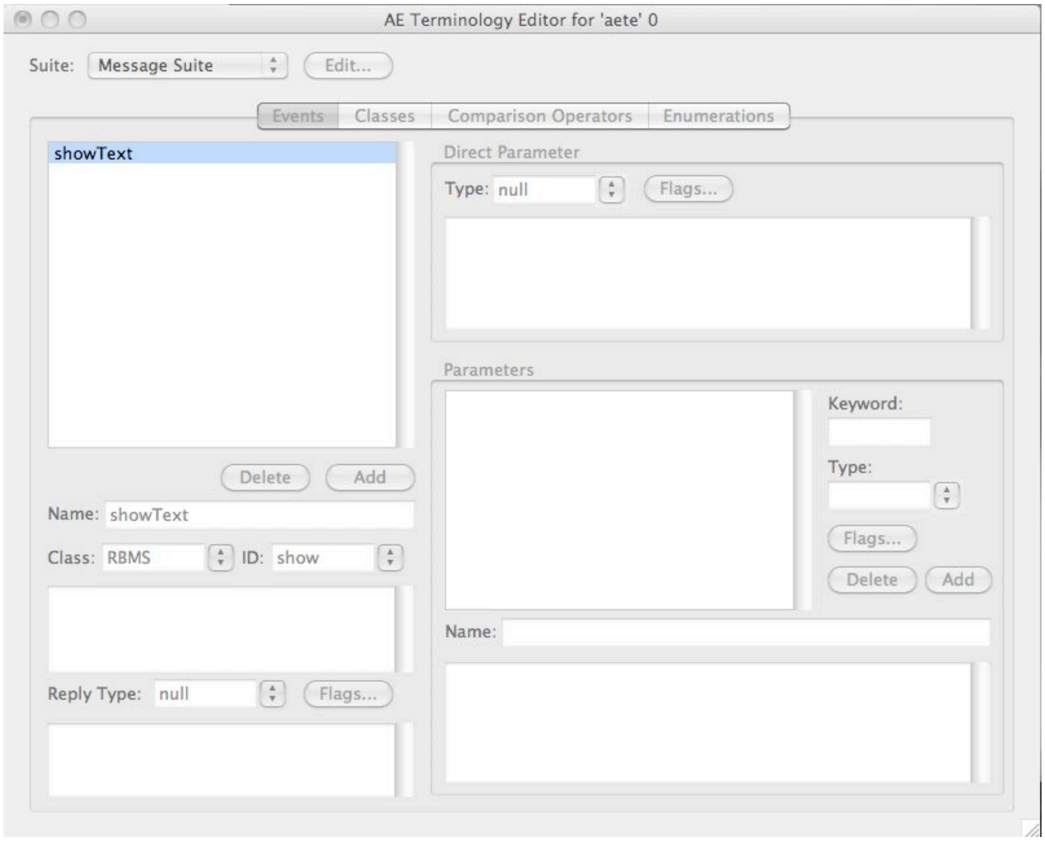
I did NOT add the entry into the .plist. It worked without it. But for a final build I will add it in to be more Apple compliant.
<key>NSAppleScriptEnabled as Boolean = "YES"</key>

Just right click on the app, and select Show Package Contents
Belongs in the root of the file.
Here is an example:

Done in DashCode, part of the monstrous X Code package.



```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist SYSTEM
"file://localhost/System/Library/DTDs/PropertyList.dtd">
<plist version="0.9">
<dict>
   <key>CFBundleExecutable</key>
   <string>RBASdemo</string>
   <key>CFBundleName</key>
   <string>RBASdemo</string>
   <key>CFBundleIdentifier</key>
   <string></string>
   <key>CFBundleInfoDictionaryVersion</key>
   <string>6.0</string>
   <key>CFBundleVersion</key>
   <string>1.0.0.0.0</string>
   <key>CFBundleDevelopmentRegion</key>
   <string>en</string>
   <key>CFBundlePackageType</key>
   <string>APPL</string>
   <key>CFBundleSignature</key>
   <string></string>
   <key>CFBundleGetInfoString</key>
   <string></string>
   <key>CFBundleShortVersionString</key>
   <string></string>
   <key>CFBundleIconFile</key>
   <string>RBASdemo.icns</string>
   <key>CFBundleDocumentTypes</key>
   <array>
   </array>
</dict>
<key>NSAppleScriptEnabled as Boolean = "YES"</key>
</plist>
```



AE Terminology Editor for 'aete' 0

Suite: Message Suite

showText

Name: showText

Class: RBMS ID: show

Reply Type: null

Direct Parameter Type: null