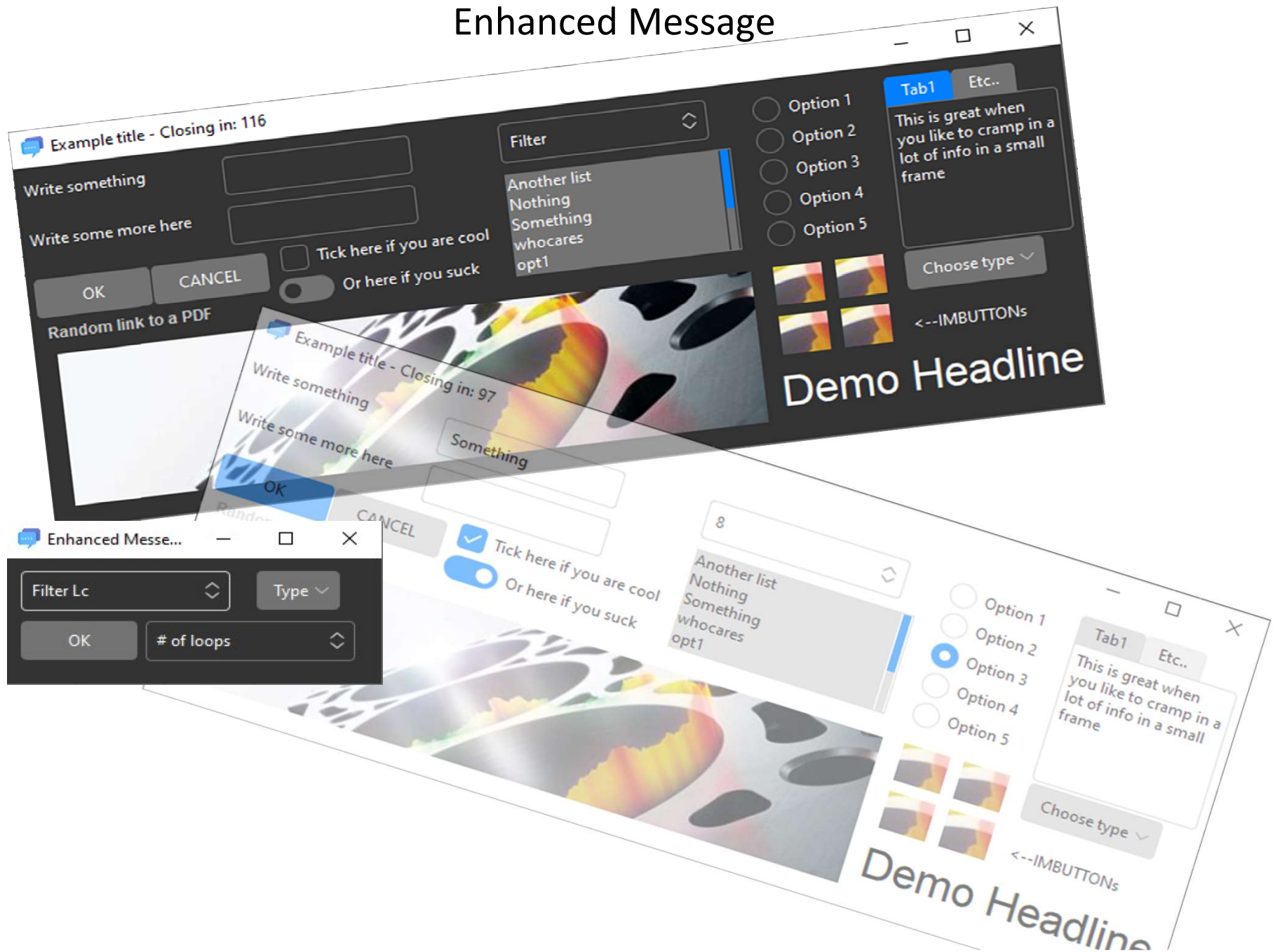


Enhanced Message





Available funcs	Required args					
	[]=optional					
TITLE	String					
IMAGE	String	Integer	Integer			
BUTTON	String	Integer	Integer			
IMBUTTON	String	String	Integer	Integer	[U D L R]	
TEXT	String	Integer	Integer			
TABBOX	List	Integer	Integer	Integer	Integer	
HEADER	String	Integer	Integer			
INPUT	Integer	Integer				
VALUE	String	Integer	Integer	Integer	Integer	Integer
SIZE	Integer	Integer				
TIMEOUT	Integer					
DROPD	List	Integer	Integer			
FORCED						
CHECK	String	Integer	Integer			
TOGGLE	String	Integer	Integer			
RADIO	List	Integer	Integer			
LIST	List	Integer	Integer	Integer	Integer	
PLAY	String					
MINIMIZED						
IMLINK	String	String	Integer	Integer		
LINK	String	String	Integer	Integer		
DARKTHEME						
FIXED						
OPEN	String [or int]	[Integer]				
MAX						
TRANSPARENT	FLOAT					
FULLSCREEN						
REDIRECT*						

TITLE	Title sets the main window title
Syntax	TITLE,String,Xoffset,Yoffset
Return value	None
Example	TITLE,"Hello world"
SIZE	Determines the size of the window
Syntax	SIZE,widht,height
Return value	None
Example	SIZE,100,200

IMAGE	You can display an un-limited amount of images. Many formats are supported, jpg, png, bmp, etc.
Syntax	IMAGE,URL,Xoffset,Yoffset
Return value	None
Example	IMAGE,path/to/img.png,10,10

BUTTON	Buttons can have any function you wish. The string arg sets the function. Un-limited amount can be used.
Syntax	BUTTON,Function,Xoffset,Yoffset
Return value	Function , in the example below, the return value will be OK
Example	BUTTON,OK,10,10

IMBUTTON	Clickable image that function the same att BUTTON. Un-limited amount IMBUTTONs can be used Optional arg can be used to animate image movement, up, down left or right.
Syntax	IMBUTTON,image,function,Xoffset,Yoffset[,U or D or L or R]
Return value	Function , in the examle below it will be Morgan.
Example	IMBUTTON,c:/img.png,Morgan,10,10,U

TEXT	Text labels. Unlimited amount of text labels can be used
Syntax	TEXT,Message,Xoffset,Yoffset
Return value	None
Example	TEXT,"Hello world",10,10

TABBOX	Text frame with different tabs. Un-limited amount of tabboxes can be used.
Syntax	TABBOX,List,width,height in rows,Xoffset,Yoffset
Return value	None
Example	TABBOX,(Tab1;info),("Another tab";blabla),10,10,30,100

HEADER	Headline, works the same as TEXT, but is bigger.
Syntax	HEADER,Message,Xoffset,Yoffset
Return value	None
Example	HEADER,cyclops,10,10

INPUT	Inputs accepts both characters and numbers as input. Un-limited amount of inputs can be used.
Syntax	INPUT,Xoffset,Yoffset
Return value	User input or EMPTY if no input is made.
Example	INPUT,10,10

VALUE	Spinbox with pre-assigned value range.
Syntax	VALUE,Headline,from,to,increment,Xoffset,Yoffset
Return value	User selecton or if no selection is made, the headline will be returned
Example	VALUE,"Choose Lc",0,10,0.5,10,10

TIMEOUT Syntax Return value Example	Can be used to automatically close the message window after X seconds. TIMEOUT,Seconds If message is timed out, return value will be timedout TIMEOUT,10
DROPD Syntax Return value Example	Creates a dropdown menu, un-limited amount of dropdowns can be used. DROPD,(Headline;opt1;opt2;etc..),Xoffset,Yoffset User selection or if no selection is made, the headline will be returned DROPD,("Favourite Beer?";Guinness;Corona;Bud;"Any beer is a good beer"),10,10
FORCED Syntax Return value Example	Removes the ability to close the message. (Make sure to use with either TIMEOUT or BUTTON. Or else you might get stuck.) FORCED None FORCED
RADIO Syntax Return value Example	Creates a set of radio buttons, only one of all options can be selected. Un-limited amount of radio buttons can be used RADIO,List,Xoffset,Yoffset The value of the radio button selected. Or 0 if none is selected. RADIO,(Opt1;Opt2;Opt3;Opt4),10,10
CHECK Syntax Return value Example	Creates checkbuttons, an un-limited amount of checkbuttons can be used. CHECK,Message,Xoffset,Yoffset 1 if checked, 0 if un-checked CHECK,"Click me",10,10
TOGGLE Syntax Return value Example	Creates a switch button, it is the same as CHECK but styled differently. TOGGLE,Message,Xoffset,Yoffset 1 if checked, 0 if un-checked TOGGLE,"Click me",10,10
LIST Syntax Return value Example	Creates a listbox that users can make multiple choices from. Only one list can be used. LIST,(opt1;opt2;etc),Box size in rows,Box width in characters,Xoffset,Yoffset User selection(s) - formatted as an array, if no selection is made, an empty array is returned. LIST,(Sweden;Finland;Norway;Iceland;Denmark),3,20,10,10
PLAY Syntax Return value Example	Plays a audio clip. Can be mp3 or WAV (other formats are un-tested) PLAY,URL None PLAY,c:/../clip.mp3

MINIMIZED	Minimizes the window. (Great if only used to play audio)
Syntax	MINIMIZED
Return value	None
Example	MINIMIZED

IMLINK	Creates a image with a hyperlink to a file, if clicked the file will be open with the default program
Syntax	LINK,IMG,URL,Xoffset,Yoffset
Return value	None
Example	IMLINK,c:/image.png,c:/file.pfd,10,10

LINK	Creates a hyperlink to a file, if clicked the file will be open with the default program
Syntax	LINK,Text,URL,Xoffset,Yoffset
Return value	None
Example	LINK,c:/../file.pdf,10,10

DARKTHEME	Applies darktheme on the message window, if not used, regular theme is applied
Syntax	DARKTHEME
Return value	None
Example	DARKTHEME

FIXED	Makes the window size fixed. Users are not able to resize the window.
Syntax	FIXED
Return value	None
Example	FIXED

OPEN	Option to specify where the prompt should open, C =center of screen, M =at current mouse pos, or at given screen coordinates
Syntax	OPEN,opt or OPEN,x,y
Return value	None
Example	OPEN,C OPEN,M OPEN,200,100

MAX	Open window maximized
Syntax	MAX
Return value	None
Example	MAX

FULLSCREEN	Open window fullscreen. No title bar will be shown. Make sure there is a way to exit the window, Eg. BUTTON etc.
Syntax	FULLSCREEN
Return value	None
Example	FULLSCREEN

TRANSPARENT	Sets the windows opacity level. Between 1-0
Syntax	TRANSPARENT,FLOAT
Return value	None
Example	TRANSPARENT,0.85

Returns

All offset values (X & Y) are pixels from top left corner

If the dialog is closed instead of clicking the intended buttons, the return value will be ERROR

The return order of the input values will always be like this (regardless of order when creating the message):

- 1) BUTTON
- 2) INPUTs
- 3) VALUEs
- 3) DROPDs
- 4) CHECKs/TOGGLEs
- 5) RADIOs
- 6) LISTs

If you have 1 TOGGLE, 3 CHECK, 1 BUTTON, 2 INPUT, 1 RADIO, 1 DROPD & 1 LIST, your order will be like this:

\$(BUTTON, INPUT1, INPUT2, DROPD, TOGGLE, CHECK1, CHECK2, CHECK3, RADIO1, \$(LIST))

Functions are separated with a white space. Therefore if a string argument contains a space, it must be double quoted.

And every functions argument is separated by comma.

Eg: TEXT,"Hello World",10,10 TEXT,"Hello again",10,100

List arguments: (x;x;x) every option is separated by semicolon and closed by parenthesis

Eg: LIST,(x;x),x,x,x,x or DROPD,(x;x;x),x,x or RADIO,(x;x),x,x

Usage:

//First, create your message window and store the return value in a variable

```
response=executeCode("WinProcess shOneOEM: '%desktop%\em.exe SIZE,531,290 IMAGE,%desktop%\pictures\pic.png,10,150  
TITLE,"Example title" TIMEOUT,120 TEXT,"Write something",5,10 TEXT,"Write some more here",5,50 INPUT,150,10 INPUT,150,50 BUTTON,OK,5,90  
BUTTON,CANCEL,90,90, DROPD,("Choose type";Staropramen;Guinness;Corona;"Any cheep beer will do!"),350,10 LIST,("Another list";Nothing  
;Something;whocares;opt1;opt2;opt3;opt4;opt5;opt6;opt7),5,27,350,50 CHECK,"Tick here if you are cool",180,83 TOGGLE,"Or here if you suck",180  
,113 LINK,"Random link to a PDF",c:/pdfs/johan.pdf,10,125 LINK,"Another link to a .png",%desktop%\pictures\rtp.png,10,145  
DARKTHEME")
```

//The returning value is formatted to be used along with ".asArrayOfLines"

```
response=response.asArrayOfLines
```

//Now all values are easily obtainable

```
for i = 1 to response.size
```

```
    display(getParameterNamed(response,i))
```

```
next i
```

//.asArrayOfLines can of course be applied directly.

```
response=executeCode("WinProcess shOneOEM: '%de....").asArrayOfLines
```

```
//A simple example to assign variables from input:
repeat
    response=executeCode("WinProcess shOneOEM: '%desktop%\em.exe SIZE,531,290 IMAGE,%desktop%\pic.png,10,150
    TITLE,"Example title" TIMEOUT,120 TEXT,"Write something",5,10 TEXT,"Write some more here",5,50 INPUT,150,10 INPUT,150,50
    BUTTON,OK,5,90 BUTTON,CANCEL,90,90, DROPD,("Choose type";Staropramen;Guinness;Corona;"Any cheep beer will do!"),350
    ,10 LIST,("Another list";Nothing;Something;whocares;opt1;opt2;opt3;opt4;opt5;opt6;opt7),5,27,350,50 CHECK,"Tick here if you
    are cool",180,83 TOGGLE,"Or here if you suck",180,113 LINK,"Random link to a PDF",c:/../johan.pdf,10,125").asArrayOfLines
until (response.first <> "timedout") and (response.first <> "ERROR")

vars=list("answer", "input1", "input2", "beerType", "list2", "iscool", "suck")
for i = 1 to vars.size-1
    setParameterNamed(getParameterNamed(vars,i),getParameterNamed(executeCode(response),i))
next i
//Lists are last since they can contain multiple indexes. Intended use:
response=executeCode(response.last) //All choises are now accessible as an array.
//Long lists (for use in LIST or DROPD) can be created by a for-loop in pcm and passed as a variable:
myList="Choise number 1"
for i = 2 to 20
    myList=myList+";Choise number "+i
next i
response=executeCode("WinProcess shOneOEM: '%desktop%\em.exe LIST,("+qm()+myList+qm()+"),10,10,10 ' ").asArrayOfLines

//Playing sounds - recommended if only used to play sounds, since it won't freeze Calypso wating for returns.
systemCall("cmd.exe /c start c:\..\em.exe MINIMIZED PLAY,c:\..\test.mp3 TIMEOUT,11")
```

***REDIRECT**

Add the redirect argument if you are redirecting the output. The output of LIST will then not be in the format of an Smalltalk array.

Output format: "op1", "op2", "opt3"....

Alternative method for indexing out put:

```
systemCallWithWait("cmd.exe /c %desktop%\em.exe LIST,(opt1;opt2),5,20,0,0 REDIRECT BUTTON,"OK",10,100 > %desktop%\out.txt")
indexedList=compute("list("+readListFile("c:\..\out.txt").last+"")")
```

To be used when runtime compiling isn't an option. General principle can also be adapted to other software.