MATLABconnector DLL Library

Project: a dynamic link library that can be called from within the MATLAB environment to make calls into the Windows WINAPI. The library needs to be called from within an 'm' file to simulate arrow key presses into another application.

Technology: the dynamic link library must be written within the 'C' language, in order for it to be importable by MATLAB (extern 'C' {} functionality does not seem to be enough). Calls must be made into the WINAPI to simulate WM_KEYDOWN (and for proper operation and simulation of actual key presses, corresponding WM_KEYUP pairs) messages being sent to the receiving application.

> Notes: SendMessage() function only seems to succeed with WM_KEYUP messages, not WM_KEYDOWN when using VK_LEFT etc key params. PostMessage() succeeds for both WM_KEYDOWN and WM_KEYUP messages.
>
> HWND_BROADCAST can be used as the HWND parameter in SendMessage()/PostMessage() to send the WM_KEY* message to all top-level windows.

Installation & Mode of Operation:

Files: MATLABconnector.dll
MATLABconnector.h
example.m
(Recipient application)

example.m needs to be copied into the MATLAB working directory, and customised to include the data feed. Each time a key press pair is required to be sent, MATLAB must call the function: *calllib ('MATLABconnector', 'fnMATLABconnector', n)* , where *n* corresponds to the relevant arrow key to be pressed:

1	LEFT ARROW
2	RIGHT ARROW
3	UP ARROW
4	DOWN ARROW

MATLABconnector.dll & MATLABconnector.h files both need to be copied into the same working directory as example.m (or added somewhere else on the MATLAB path, but for simplicity it is suggested that they are kept with the 'm' file).

Once the recipient application is running, the MATLAB m file can be run, and the key messages should be passed to the recipient application.

The library sends the WM_KEYDOWN & WM_KEYUP messages to all top-level applications, so it is not dependent on finding the recipient application directly. As all applications will receive these messages, it is advised not to have any unnecessary applications running (e.g. word processing tools etc) as the cursor will be moved around a lot! Whilst these are harmless messages in most contexts, this does not take into account every scenario. If further customisation of the library is required to focus the messages on a single recipient application, this could be achieved relatively easily.

KeysTestApp: C# test application

A very quick and simple test application to demonstrate that the WM_KEYDOWN and WM_KEYUP messages are processed; this application needs to be launched in Debug mode through Visual Studio, as it outputs its messages to the 'Output' Console through the Visual Studio IDE. Other than 'proof of concept' this application has no relevance to the operation of the main dll or Matlab files, and is not required for successful operation. * important: Not all functionality within this application has been completed!

Source:

Source code is provided for both the dll and for the test application. These projects are built in Visual Studio 2005, and do not require any additional libraries or dependencies beyond what is provided with Visual Studio.

Matlab code has been developed using MATLAB 7 R14, but should be compatible with similar versions of Matlab.