Mainframe Automation

Blue Prism Mainframe Interface

Blue Prism has a built in interfaces to expose the API functionality with a wide variety of mainframe emulators.

Launching

IBM and Hummingbird mainframe sessions are launched using the details contained in the host session file. This session host file contains all the configurable details of the session such as screen size, cursor type etc.

For all sessions you can simply start the session. The exception is PowerTerm that requires the session to already be running (see Vendor Specific Details)

Session Identifier

You must specify a session identifier when starting or attaching to a mainframe session. This can be hardcoded in Application Modeller or set dynamically via parameters on the Launch and Attach stages.

If you intend to run multiple concurrent sessions on a single Blue Prism resource it is best to environment locks to allocate session identifiers rather than hard code them to specific processes. This allows you to run multiple sessions on the same resource without risk of collision.

With the exception of Attachmate you can attach to any existing session simply by providing the session identifier. This enables you break up you object layer and have multiple objects supporting the mainframe application.

Spying Elements

The mainframe session supported within Blue Prism expose an API that allows Blue Prism to

Blue Prism provides a grid tool to allow you to quickly spy regions on the mainframe screen. For details on how to use this see the "Guide to Using the Blue Prism Mainframe Spy".

Once an element has been captured it is possible to copy and paste that element within Application Modeller. This allows you to create new elements quickly and simply rename them and set the appropriate X, Y coordinates. This can be useful if you already know the screen coordinates of the field you want to interact with or the coordinates are displayed on the mainframe screen.

When spying fields within a mainframe session it is not always possible to determine where the field ends. The following screen shot displays a mainframe screen where the field sizes are known.

```
                    SAMPLE APPLICATION FORM

APPLICATION NO :  _____

                  ------------------------------------------
                  |READ DETAILED INSTRUCTIONS GIVEN SEPARATELY|
                  |    BEFORE FILLING THE APPLICATION FORM.    |
                  ------------------------------------------

  NAME OF THE APPLICANT :  _____   _____  _____
                           FIRSTNAME  MIDDLE      LAST-NAME
  DATE OF BIRTH : __ / __ / ____
  RESIDENTIAL ADDRESS :  _____
                         _____

  EDUCATIONAL DETAILS
  ------------------------------------------
  |QUALIFICATION | UNIVERSITY    | YEAR    |
  ------------------------------------------
  | _____     | _____      | _____   |
  | _____     | _____      | _____   |
  | _____     | _____      | _____   |
  ------------------------------------------
```

Here you can easily specify the field widths for all the fields as it is indicated by the green line.

On the following mainframe screen the input fields are not easily identifiable.

```
                    SAMPLE APPLICATION FORM

APPLICATION NO :

                  ------------------------------------------
                  |READ DETAILED INSTRUCTIONS GIVEN SEPARATELY|
                  |    BEFORE FILLING THE APPLICATION FORM.    |
                  ------------------------------------------

  NAME OF THE APPLICANT :
                           FIRSTNAME  MIDDLE      LAST-NAME
  DATE OF BIRTH :      /      /
  RESIDENTIAL ADDRESS :

  EDUCATIONAL DETAILS
  ------------------------------------------
  |QUALIFICATION | UNIVERSITY    | YEAR    |
  ------------------------------------------
  |              |               |         |
  |              |               |         |
  |              |               |         |
  ------------------------------------------
```

If we are inputting text to the field we can use the cursor and type text until we reach the end of the field i.e. populate all the available space in all the fields on the screen.

However if we are reading from the screen we won't know the sizes of the fields. For example where does the residential address field end? In these situations we identify the field to be the maximum width available on the screen to avoid the risk of cutting off the end of field data as below.

```
                    SAMPLE APPLICATION FORM

APPLICATION NO :

                  -------------------------------------------
                  |READ DETAILED INSTRUCTIONS GIVEN SEPARATELY|
                  |    BEFORE FILLING THE APPLICATION FORM.   |
                  -------------------------------------------

NAME OF THE APPLICANT :  [        ] [        ] [            ]
                          FIRSTNAME  MIDDLE     LAST-NAME
DATE OF BIRTH :[    ]/[    ]/[    ]

RESIDENTIAL ADDRESS :[                                      ]
                     [                                      ]
EDUCATIONAL DETAILS  [                                      ]
-----------------------------------------------------
|QUALIFICATION | UNIVERSITY      | YEAR    |
-----------------------------------------------------
|[          ] | [            ]  | [      ] |
|[          ] | [            ]  | [      ] |
|[          ] | [            ]  | [      ] |
-----------------------------------------------------
```

Trimming Data

Whether the field sizes are known or not what is essential when reading from a mainframe session is that you trim all data once you have read it from the screen. This will remove any leading or trailing spaces. Typically you will read from the screen using one read stage then trim all data using a Multiple Calculation stage before casting any dates or numbers.

Vendor-Specific Details

IBM

A session can be started from Object Studio by specifying in the Application Modeller a path to the session file.

For IBM Personal Communications Blue Prism has built into the interface the Wait for Idle host command which follows and sending of keys to the emulator. As a result you will not need arbitrary waits following a send key as Blue Prism will only pass control back to the process when the emulator is ready.

When using IBM Personal Communications you set screen text (where the host application allows) by simply using a Blue prism Write stage to send the required text to the specific coordinates of the screen. To send non alphanumeric keys you must use a Global Send Key within a navigate stage and send the correct Mnemonic keyword.

Mnemonic keywords

Popular send keys Mnemonic keywords:

Keyword	Description

[enter]	Enter
[delete]	Delete
[tab]	Tab
[clear]	Clear Screen
[attn]	Escape or Attention
[pf1]	Function keys [pf {key number}]

For a full set of all keywords please see Appendix A

Powerterm

(Applies to version 9.2)

Launching and Attaching

Open your session in Powerterm, set up as desired and save as a "pts" file. The session letter is not contained in this file and the session letter cannot be configured from the application's setup menus.

Use a code stage to launch the ptw32.exe file, passing the path to the session file as a command line argument. The code stage will wait for the application to be launched.

Use a regex to extract the short session letter from the window title of the application's main window.

Pass the session letter out of your code stage to use as an input to the "launch" action of your HLLAPI Powerterm business object. This will attach to the running session.

Mnemonic keywords

Popular send keys Mnemonic keywords (see Appendix B for full list of Attachmate mnemonics):

Keyword	Description
@E	Enter
@T	Tab
@1	Function key F1 (@ {function key). See the "keyboard map" under the Powerterm application menu to confirm mapping of function keys.

Toubleshooting

The specified module could not be found

Launching a mainframe app causes an error message, similar to : "Unable to load DLL 'HLLAPI32.DLL': The specified module could not be found"

This is due to the mainframe installation not adding its install directory to the Windows PATH environment variable.

To resolve this issue, open System Properties in the Windows Control Panel, enter Advanced, click the Environment Variables button and append the path to the directory containing the referenced file to the PATH variable, prefixed with a semi-colon.

My mainframe app exits when my process finishes / resets

When a Blue Prism Visual Business Object accesses a mainframe, it opens a connection through a terminal emulator onto that mainframe, and processes any communication through that emulator.

When the process using the VBO is completed or reset, the VBO closes the connection it opened within the terminal emulator.

Some emulators will leave the mainframe connection and corresponding window open, but some (specifically some EHLLAPI emulators) will close themselves down, severing the connection to the mainframe as part of their disconnection process. This could potentially leave users logged into the mainframe session if the process has not logged them out normally.

One way around this is to launch the emulator separately and attach to it using the appropriate action within the VBO. This ensures that the mainframe session is not terminated when the VBO is reset.

Alternatively, care should be taken to ensure that the session is logged out in the Clean Up page of the VBO.

Appendix A – IBM Mnemonic keywords

Host key	Mnemonic	Host key	Mnemonic
Back tab	[backtab]	PF18	[pf18]
Clear screen	[clear]	PF19	[pf19]
Delete	[delete]	PF20	[pf20]
Enter	[enter]	PF21	[pf21]
Insert	[insert]	PF22	[pf22]
Jump	[jump]	PF23	[pf23]
Left	[left]	PF24	[pf24]
New line	[newline]	End of file	[eof]
Space	[space]	Scroll Lock	[scrlock]
Print	[print]	Num Lock	[numlock]
Reset	[reset]	Page Up	[pageup]
Tab	[tab]	Page Down	[pagedn]
Up	[up]	PA 1	[pa1]
Down	[Down]	PA 2	[pa2]
DBCS	[dbcs]	PA 3	[pa3]
CapsLock	[capslock]	Word Delete	[worddel]
Right	[right]	Field Exit	[fldext]
Home	[home]	Insert Toggle	[instog]
PF2	[pf1]	Cursor Select	[crsel]
PF2	[pf2]	Cursor Left Fast	[fastleft]
PF3	[pf3]	Attention (escape)	[attn]
PF4	[pf4]	Print Presentation Space	[printps]
PF5	[pf5]	Cursor Up Fast	[fastup]
PF6	[pf6]	Cursor Down Fast	[fastdown]
PF7	[pf7]	Cursor Right Fast	[fastright]
PF8	[pf8]	Underscore	[underscr]
PF9	[pf9]	Print (PC)	[printpc]
PF10	[pf10]	Forward Word Tab	[wordright]
PF11	[pf11]	Backward Word Tab	[wordleft]
PF12	[pf12]	Field -	[field-]
PF13	[pf13]	Field +	[field+]
PF14	[pf14]	Print Presentation Space on Host	[printhost]
PF15	[pf15]	Alternate Cursor	[altcsr]
PF16	[pf16]	Backspace	[backspace]
PF17	[pf17]		

Appendix B - Attachmate Mnemonic keywords

Host key	Mnemonic	Host key	Mnemonic
@	@@	Home	@0
Alternate Cursor	@$	Insert	@I
Attention	@A@Q	Jump	@J
Backspace	@<	New Line	@N
Backtab	@B	Num Lock	@t
Blue	@A@h	Page Down	@v
Caps Lock	@Y	Page Up	@u
Clear	@C	PA1	@x
Cursor Down	@V	PA2	@y
Cursor Left	@L	PA3	@z
Cursor Left Double	@A@L	PF1	@1
Cursor Right	@Z	PF2	@2
Cursor Right Double	@A@Z	PF3	@3
Cursor Select	@A@J	PF4	@4
Cursor Up	@U	PF5	@5
Delete	@D	PF6	@6
Delete Word	@A@D	PF7	@7
Device Cancel	@A@R	PF8	@8
DUP	@S@x	PF9	@9
End	@q	PF10	@a
Enter	@E	PF11	@b
Erase to EOF	@F	PF12	@c
Erase Input	@A@F	PF13	@d
Reset Reverse Video	@A@c	PF14	@e
Field Mark	@S@y	PF15	@f
Green	@A@f	PF16	@g
Reset Host Colors	@A@l	PF17	@h
Reverse Video On	@A@9	PF18	@i
Scr Lock	@s	PF19	@j
System Request	@A@H	PF20	@k
Tab	@T	PF21	@l
Test	@A@C	PF22	@m
Turquoise	@A@i	PF23	@n
Underscore	@A@b	PF24	@o
White	@A@j	Pink	@A@e
Word Tab Back	@A@z	Print PS	@A@T
Word Tab Forward	@A@y	Print Screen	@P
Yellow	@A@g	Queue Overrun	@/
(reserved)	@X	Red	@A@d
Reset	@R	Field Exit	@A@E
Cursor Up Double	@A@U	Cursor Down Double	@A