

MACRO: Frontlog 2006-02 (draft)

NOTE: The macro was created by Jared Schwede (student working for SML Acquisitions) in 2004 and rev. and re-named by Eric Friede in 2006 to account for changes in Voyager 5.0.

BACKGROUND. When cataloging copy is not available at the time of receipt, SML Acquisitions staff route items without copy to the Frontlog, a caged area in the SML basement containing some 100,000+ items. For status tracking purposes, the physical item is associated with Orbis with an item record, and the item record is associated with a 14 digit barcode number, e.g. 39002059419508. The barcode itself is affixed to the book (or other physical item equivalent) and charged to a dummy Orbis patron, e.g. frontlg245.

Although there is a maximum number of potential charges to a frontlog patron, that number is large enough to warrant further subarrangement to facilitate retrieval due to patron requests. The subarrangement is based on an additional 6 digit barcode number printed on a slip kept with the book. The "Frontlog number" always begins with UNCAT followed by 6 digits, e.g. UNCAT225565. This number is used to shelve the book in the FRONTLOG area in the SML basement. In order for Circulation staff to find the book for a patron request, the number is entered in the Orbis MFHD record field 852 \$h. This requires SML Acquisitions staff to change the MFHD 852 \$h from this:

8	‡b sml ‡h In
5	Process
2	

to this:

8	‡b sml ‡h
5	UNCAT225565
2	

OPERATION (General). The user sets up by opening the Cataloging and Circulation modules. From the Circulation module, the user opens the Charge window and enters the dummy patron name. If any windows are open in the Cataloging module, the macro is started from Cataloging to ensure that all Cataloging module windows are closed. (??)The user then turns on the macro, usually mapped to CTRL+f.

- The macro first prompts the user for the barcode number with a dialog box.
- When the barcode number is entered, the macro charges the item to the Frontlog status (the macro will override any intervening Circulation windows to do this).
- The macro then switches Orbis from the Circulation module to the Cataloging module,
- retrieves the item record via the item barcode number,
- opens the associated MFHD record,
- selects the In Process text in 852 \$h, and
- opens a dialog box prompting the user for the frontlog barcode number.
- When the user wands the barcode number from the slip, the selected In Process text is replaced by the barcode number.
- The macro then switches Orbis from the Cataloging Module back to the Circulation and
- opens the dialog box prompting the user for the next item barcode number.

MACRO SCRIPT COMMENTS

1. Macro Playback speed: Normal. Because this is a complicated macro, the Macro Playback speed is included to allow the option of a slower playback if the macro needs to be tweaked.

To set Playback Speed:

- Select Timing from the Commands window,

- Double-click on Macro Speed to open the Speed dialog box,
- Click on the Macro Speed button,
- Enter Playback Factor 1 for normal playback speed.
- The macro script line should read: *Macro Playback Speed: Normal Speed*

2. If condition. The macro is intended to work with no record windows open in the Cataloging module, so this series of commands is intended to ensure that all record windows in the Cataloging module are closed. This is especially important since the macro needs to open some Cataloging windows later in the sequence, and these windows need to be closed before the macro can repeat. (??) If no Cataloging module windows are open, the macro will ignore this sequence (2.) and go on to 3 (but you don't need to specify this in the macro script).

- Select **Logic** from the **Commands** window,
- Double-click on the **If Window** option to open the If Window dialog box,
- From the Commands list in the If Window dialog box, click on the radio button **If Window on Top**
- In the same If Window dialog box, find below the Commands options list the **Window Title:** prompt slot, and enter the text: *Voyager Cataloging - [*
- Leave the **Partial Match** radio button selected
- Click **OK** button
- The macro script line should read: *If Window Title "Voyager Cataloging - [" is on top*

Comment: The window title *Voyager Cataloging - [* ensures that the series of commands operates only if the Voyager Cataloging window is open. The **Partial Match** option means that the commands will work even if a specific record number and title are displaying after the initial *Voyager Cataloging - [* on the title bar of the Cataloging Module window, e.g. *Voyager Cataloging - [Bib 6018213 : Sobre o vale do mosteiro / Eugenio Mendes Pinto.]*

- Select **Text** from the Commands window,
- Double-click on the **Text Type** option to open the Text Type window
- Below the blank **Keystrokes** space, several columns of function/command keys are listed. Select (single-click) **Control** in this list, which will cause the text *<CONTROL>* to appear in the Keystrokes space.
- Type the letter **q** after *<CONTROL>*; the keystroke sequence is now: *<CONTROL>q*
- Click the **OK** button
- The macro script line should read: *Text type: <ctrl>q*

Comment: In Voyager, the key sequence *<ctrl>q* saves the record. The following macro command sequence accounts for the dialog box that follows a save (*<ctrl>q*) command in Voyager, requiring the user to press ENTER to complete the save, followed immediately by the command sequence to close all open windows.

- Select **Timing** in the Commands window,
- Double-click on **Wait Window Lose Focus** to open the Timing Commands dialog box, where that option will have been selected,
- In the **Window Title** prompt slot, enter *Voyager Cataloging - [* (or use the browser [ellipses] button to find & select the title),
- Click on the **Wait a Maximum of:** radio button (the current macro uses 10 minutes), and,
- Click **OK**
- The macro script line should read: *Wait Window Lose Focus: "Voyager Cataloging - ["*
- Select **Text** from the Commands window,
- Double-click on **Text Type** option to open the Text Type window
- From the command keys list under the blank **Keystrokes** space, click on **Enter** to select the *<ENTER>* key, then click on **Alt** to select the *<ALT>* key, then
- Type *fo*; the line should now read: *<ENTER><ALT>fo*
- Click **OK**
- The macro script line should read: *Text Type: <ENTER><ALT>fo*

Comment: *<ALT>f* opens the Voyager **File menu**; *o* selects the **Close All** command from the menu. The "if" situation where all Cataloging module windows need to be closed is now completed. However, Macro Express

requires you to indicate that the "if" sequence is over, so:

- Select **Logic** from the Commands window,
- Double-click on the **End If** option (*End If* will appear as the last line of the macro script **If** sequence)

3. Now that all the windows in the Cataloging module are closed, the macro needs to activate the Circulation module window. Before this is done, the Circulation window is maximized.

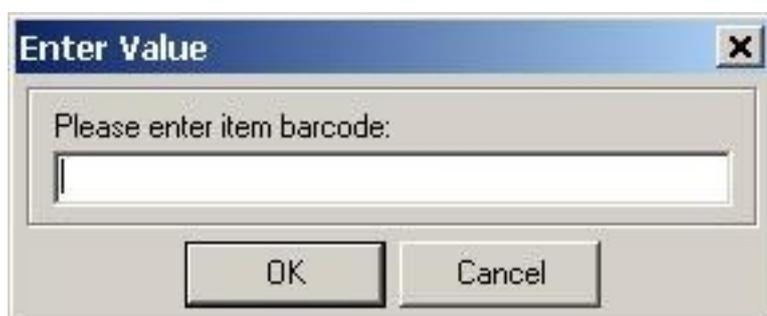
- Select **Windows/Programs** from the **Commands** window.
- Double-click on **Windows Maximize** to open the **Windows Commands** dialog box; the **Maximize Windows** radio button should be pre-selected under the **Commands** options.
- Under the **Window** options, click on the **Specific Window** radio button to select that option and type *Voyager Circulation* in the option slot
- The macro script line should read: *Window Maximize: "Voyager Circulation"*

Comment: Maximizing the Circulation window may no longer be necessary. The original version of this macro depended on mouse positioning, which worked more effectively with the Circulation module window maximized. Macros based on mouse positions are high-maintenance, since the positions need to be recalculated for different workstations or after software upgrades. The current version is based on key sequences (keyboard shortcuts). Key sequences don't change from workstation to workstation, and usually the only problem is if a key sequence or the screen layout changes after a software upgrade (as has been the case at times with Voyager 5.0).

4. Now the Circulation window must be activated, since you want to perform a function (charging the item barcode) in the Circulation module:

- Select **Windows/Programs** from the **Commands** window.
- Double-click on **Window Activate** to open an abbreviated version of the **Activate/Launch Program** dialog box. The **Activate Window** radio button should be pre-selected under the **Activate/Launch Program Commands**.
- In the **Activate Window** option slot, type *Voyager Circulation*, then click OK.
- The macro script should read: *Activate Window: "Voyager Circulation"*

5. At this point, we want the macro to open a dialog box to prompt the user to wand in the item barcode number. The prompt dialog box will look like this:



As an insurance measure, just to make sure that another variable has not been stored in the memory, a line of code is entered to clear variables before the prompt dialog box is evoked.

- Select **Variables** from the **Commands** window.
- Double-click on **Clear Variables**; the defaults should be **Variable Type to Clear: Text Variables** with radio button **Clear All** selected.
- Click **OK**; the macro script line should read *Clear Text Variables: All*

To create the prompt dialog box:

- Select **Variables** from the **Commands** window.
- Double-click on the **Variable Set String** option to open the **Set Text String Variables** dialog box.
- Under **Commands** in this dialog box, click on the radio button **Prompt for Value**.
- **Variable Name** *T1* is the default, which is OK since this will be the first text variable in this macro.
- The **Prompt Text** field will open within the Set Text String dialog box. Type in: *Please enter item barcode*
- The center/center default positioning of the prompt box is OK to leave as is.
- Click OK; the macro script should read: *Variable Set String %T1% from Prompt*

6. The expected response to Please enter item barcode is for the user to wand the item barcode (i.e., the T1 variable text string). A delay of 100 milliseconds is added to the script to provide sufficient time for the macro to tell the computer to read and record in memory the wanded item barcode number and close the Please enter item barcode prompt window. With the Circulation module Charge Window now on top, the macro is then instructed to type the T1 variable (the item barcode number it has just read); the only space available for the number to be typed is the BARCODE slot.

7. With the item barcode number now typed into the BARCODE slot, the macro can now ENTER the number (text) string. A delay of 100 milliseconds is added to give Orbis time to bring up the Item Blocks and Information Window, an unfortunate extra step built into the Voyager technical services charging procedure whenever a new item barcode number is entered in a Tech. Services charge. The macro runs the key sequence <ALT>o to override the window. Depending on the workstation, another Item Blocks window may open, so another If condition sequence is added. The item is now charged to the Frontlog dummy patron.

8. The Circulation window is now minimized and the Cataloging module window is activated. In the Cataloging module, the Retrieve an Item Record (<ALT>rb) prompt window is opened. A delay of 100 milliseconds is entered to give the window time to open. The item barcode number (T1 variable) is typed in by the macro, with another 100 millisecond delay for the number to be read by Voyager. The Text Type <ENTER> (the equivalent of pressing the ENTER key) tells Voyager to retrieve the item record. Another delay is added to the script to give Orbis time to retrieve the item record. <maybe not necessary, since it's followed by the Wait for Window>

9. The Wait for Window Title accounts for different response times. A key sequence <ALT>rg is used to open the MFHD associated with the item record retrieved at 8. Another delay is written in (again the Wait for Window ought to take care of variations in response time). The clear variable command must be re-executed to clear out the T1 variable (the item barcode) so it can be replaced by the T2 variable, the frontlog barcode, or, alternatively, letters to initiate one of two If conditional sequences.

10. The prompt for the T2 variable is created. The T2 can be either the frontlog barcode number or a letter to initiate one of two sub-macros for lnb and art. If the location is sml, the frontlog barcode is wanded and the process continues.

11. If the letter "l" or "a" is entered, different sub-macros are run. This requires 2 If condition sequences to cover backlog locations lnb and art, initiated by a prompted letter "l" or letter "a" requested by the prompt. If a sub-macro is run, it loops back to the original prompt at which point the user is prompted to wand in the frontlog barcode only.

12. With the frontlog barcode=T2 based on 10. or 11., another sub-macro is inserted into the script, GOTO 852 in MFHD. The function of this macro is to align the cursor with the selection bar for the 852 field in the MFHD.

13. From the selection bar, a key sequence of TABS and <CTRLD><ARROW LEFT/RIGHTs> are used to select the In Process text in 852 \$h, which is then copied to the clipboard and replaced with the T2 variable text, i.e. the frontlog barcode number. <BUT WHERE IS IT PASTED INTO THE MFHD??>

14. To account for situations where there is no In Process text or the macro did not successfully copy the text into the clipboard, an If conditional is added to halt the macro under any condition where the text is not In Process.

15. The macro is nested to allow indefinite repetition.

Source URL: <https://web.library.yale.edu/cataloging/macroexpress-frontlog>