



px.factory Version 1.2.01

User Manual

(C) Derek Cook 2019-2020

www.xfactory-librarians.co.uk

Table of Contents

<u>What's New in px.factory V1.2.01 (20/12/2022)</u>	1
<u>Previous Versions</u>	1
<u>V1.2.00 (08/05/2022)</u>	1
<u>V1.1.01 (05/07/2020)</u>	1
<u>V1.1.00 (18/04/2020)</u>	1
<u>V1.0.00 (22/12/2019)</u>	1
<u>V0.1.00 (10/12/2018)</u>	1
<u>Installation and Configuration of px.factory</u>	2
<u>Registering px.factory</u>	3
<u>Running px.factory</u>	4
<u>PC</u>	4
<u>MAC OS X</u>	4
<u>Command Line</u>	4
<u>The px.factory Window</u>	5
<u>Menu Bar</u>	6
<u>Main Menu Headings</u>	6
<u>The Application Menu</u>	7
<u>About px.factory</u>	7
<u>Preferences</u>	7
<u>Quit px.factory</u>	7
<u>The File Menu</u>	8
<u>New</u>	8
<u>Open</u>	8
<u>Save</u>	8
<u>Save As</u>	8
<u>Save All</u>	8
<u>Open Workspace</u>	8
<u>Save Workspace</u>	8
<u>Export to</u>	8
<u>Exit</u>	9
<u>Recently Opened Files</u>	9
<u>The Edit Menu</u>	10
<u>Undo</u>	10
<u>Redo</u>	10
<u>Clear Undo Buffer</u>	10
<u>Cut</u>	10
<u>Copy</u>	10
<u>Paste</u>	10
<u>Find</u>	10
<u>Find Next</u>	10
<u>Goto</u>	10
<u>The View Menu</u>	11
<u>Tool Bar</u>	11
<u>Status Bar</u>	11
<u>No Split</u>	11
<u>Horizontal Split</u>	11
<u>Vertical Split</u>	11
<u>Tree</u>	11
<u>Table</u>	12
<u>Refresh</u>	12
<u>Preferences</u>	12
<u>The Window Menu</u>	13
<u>Tile Horizontal</u>	13
<u>Tile Vertical</u>	13
<u>Maximise Windows</u>	13
<u>Restore Windows</u>	13
<u>Minimise Windows</u>	13
<u>View Prophet X Synthesizer Data</u>	13
<u>The Help Menu</u>	14
<u>Search</u>	14
<u>Get Help on px.factory</u>	14
<u>View Tips and Tricks for px.factory</u>	14
<u>View PDF Documentation for px.factory</u>	14
<u>View License Agreement for px.factory</u>	14
<u>View License Agreement for px.factory</u>	14
<u>About px.factory</u>	14
<u>Register and Activate px.factory</u>	14

Table of Contents

<u>The px.factory Window</u>	
Tool Bar.....	15
Desktop.....	16
Child Window.....	17
Tree View.....	18
Table View.....	19
Status Bar.....	21
<u>px.factory Program Features</u>	22
File Operations.....	23
Create New File.....	24
Open Existing File.....	25
Save Open Files.....	26
Workspaces.....	27
Export Data.....	28
HTML	28
XML	28
SpreadSheetML	28
Undo Buffer.....	29
Clipboard Support.....	30
Find and Goto Facilities.....	31
Find Dialog.....	31
Goto Dialog.....	31
Editing Features.....	32
General Editing Features.....	33
Copy/Move Object.....	34
Initialise Object.....	34
Rename Object.....	34
Insert Object.....	35
Delete Object.....	35
List Object References.....	35
Sort Objects.....	36
Synth Editing Features.....	36
Import from Prophet X Memory.....	36
Export to Prophet X Memory.....	36
Program Editing Features.....	37
Program Bank Collection Editing Features.....	37
Program Bank Editing Features.....	37
Program Editing Features.....	37
Import from Prophet X Memory.....	38
Export to Prophet X Memory.....	38
Import from Prophet X Edit Buffer.....	38
Export to Prophet X Edit Buffer.....	38
Play List Editing Features.....	38
Play List Editing Features.....	38
Import from Prophet X Memory.....	38
Export to Prophet X Memory.....	38
Play List Set Editing Features.....	38
Global Editing Features.....	38
<u>Preferences</u>	40
New File Default Preferences.....	41
Play List Preferences.....	42
Check Preferences.....	43
Default View Preferences.....	44
MIDI Preferences.....	46
Miscellaneous Preferences.....	48
<u>Online Help</u>	49
Context Sensitive Help.....	49
<u>Tips and Tricks</u>	50
List of Tips and Tricks.....	50
Context Sensitive Help.....	50
Keyboard Shortcuts.....	50
Object Colouring.....	50
Find/Find Next Category Tool Bar Buttons.....	50
Table Sorting.....	50
You can only DRAG COPY Certain Objects.....	50
Copying/Moving Objects to a Different File.....	51
Workspaces.....	51

Table of Contents

<u>Tips and Tricks</u>	
<u>Moving Objects within a File</u>	51
<u>About px.factory</u>	52
<u>Glossary</u>	53
<u>Keyboard Short Cuts</u>	54
<u>Keyboard Shortcuts for the Desktop and Child Windows</u>	54
<u>Keyboard Shortcuts for Tree Views</u>	55
<u>Keyboard Shortcuts for Table Views</u>	56
<u>Keyboard Editing Short Cuts for both Tree and Table Views</u>	57
<u>Keyboard Shortcuts for the Preferences Dialog</u>	58
<u>Keyboard Shortcuts for Text Editing Fields</u>	58
<u>Credits</u>	59
<u>3rd party software used in px.factory</u>	59
<u>CoreMIDI4J License</u>	59
<u>Table Filter License</u>	59
<u>Apache Derby License</u>	59
<u>Apache Log4J License</u>	59
<u>Dom4J License</u>	59
<u>Java Wizard License</u>	59
<u>px.factory License</u>	60
<u>Disclaimer</u>	60
<u>Licensing Conditions for 3rd party software</u>	60
<u>Apache Derby License</u>	60
<u>Apache Log4J License</u>	60
<u>CoreMIDI4J License</u>	60
<u>Dom4J License</u>	60
<u>Java Wizard License</u>	61
<u>Table Filter License</u>	61
<u>Tritonus License</u>	61

What's New in px.factory V1.2.01 (20/12/2022)

- You can now double click in a table to select lower level tables rather than using the context menu. (e.g. Program Bank Collection, Program Banks) and px.factory will move down to the next level until you get to discrete objects (e.g. Program) You can double click with the ALT key (OPTION key on OS X) held down to traverse upwards
- Updated how the applications are built. The Java Runtime is now embedded and updated to Java 17 as the minimum version. There is no longer a need for a standalone version of Java to be installed
- New installers for OS X (DMG), PC (MSI) and Linux (DEB) and basic ZIP and TAR packages.
- The OS X DMG package is notarized for compatibility with the OS X security model.

Previous Versions

V1.2.00 (08/05/2022)

- Updated how the applications are built. The Java Runtime is now embedded and updated to Java 17 as the minimum version. There is no longer a need for a standalone version of Java to be installed
- New installers for OS X (DMG), PC (MSI) and Linux (DEB) and basic ZIP and TAR packages.
- The OS X DMG package is notarized for compatibility with the OS X security model.

V1.1.01 (05/07/2020)

- ◆ Fixed problem on OS X Mojave and Catalina which was preventing access to protected folders.

V1.1.00 (18/04/2020)

- ◇ The Minimum Java Version is now Java 11, and I have rebuilt the applications with the latest support libraries. The migration to Java 11 is a significant step, as I have been help back from migrating to Java 9 and above due to Java PC MIDI issues that were causing garbled MIDI port names, which made it impossible to select MIDI ports. These issues are now resolved in the recommended JREs. Please see the [Java Support](#) page on the website for more details.

- ◇ The applications support iMac Retina displays.

- ◆ The Minimum Java Version is now Java 11.
- ◆ The applications support iMac Retina displays.

V1.0.00 (22/12/2019)

- First Release

V0.1.00 (10/12/2018)

- Beta release for internal testing

Installation and Configuration of px.factory

px.factory is downloaded as a self installing file, so installation is automatic, and for most users little else, if anything, needs to be done to get up and running.

A native installer for Windows is also available.

Simply download the installer from the [x.factory Products Page/a> for your platform \(PC, OS X or Linux\) and run the installer, following the installer prompts.](#)

Registering px.factory

You need to register px.factory to allow data to be saved to your computer and for [Library Support](#) to be enabled.

Please refer to the [x.factory Librarian Registration Page](#) for details on how to register px.factory.

Running px.factory

Starting up px.factory is as simple as starting any application, simply double click on the icon and you're away.

If you have [registered](#) px.factory, then the Library will be created (if it doesn't already exist) and opened.

During start-up, a "Splash Screen" is shown. This stays visible until a few seconds after the program has loaded and is ready for use, but can be dismissed at any time by clicking on the Splash Screen.

px.factory remembers the size and position of the main window between sessions, so the main window will be sized and positioned as you left it the last time.

PC

Simply double click on the program icon for the program either in an Explorer shell, or from the desktop if you let the installer place a shortcut there.

You can also select the program from the **Start Menu**. By default, px.factory can be found under the **x.factory Librarians** Program Group.

MAC OS X

Navigate to the **Applications** folder in **Finder** and double click the px.factory application.

You can also create "**Aliases**" of the application and place them anywhere you wish, such as on the desktop

To make an alias you can choose any one of the following methods:

- Click the px.factory item you have just identified and hold down the mouse button. Then, while holding down the **Command** and **Option** keys simultaneously, drag the item to where you want the alias to appear then let go of the mouse button and finally release the two keys. This will create an alias at your desired location. While dragging, the 'ghosted' item will display a shortcut arrow which will remain on the aliased icon when it has been created.
- Hold down the **Ctrl** key, and then click the px.factory item as identified previously. The context menu will be displayed, and one of the choices will be **Make Alias**. Click that option, and then release the key. The px.factory alias will appear next to the original item and can be dragged to your location of choice.
- Select the px.factory item you identified previously by clicking it once then press **Command-I** (lowercase L). The px.factory alias will appear next to the original item and can be dragged to your location of choice.
- Click the px.factory item once to select it, and then from the **File** menu, select **Make Alias**. The px.factory alias will appear next to the original item and can be dragged to your location of choice.

Note that if your alias does not have a small arrow in the lower left corner then you have not correctly created an alias and have probably just dragged the application launcher to a new location. If so, drag it back to the Application folder and start again.

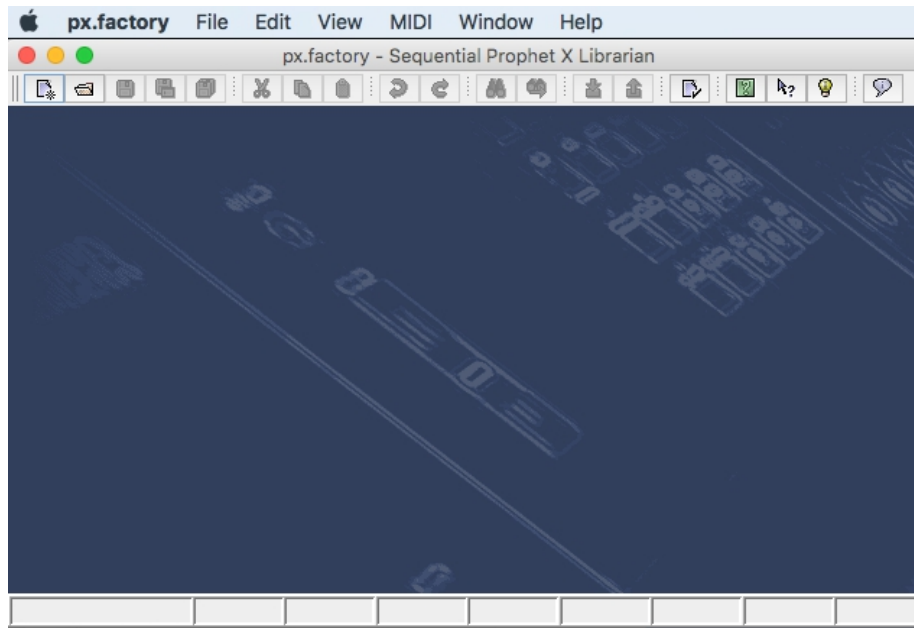
Command Line

You can also startup px.factory from a command shell (e.g. MsDOS). Simply change directory to where px.factory is located and type the following:

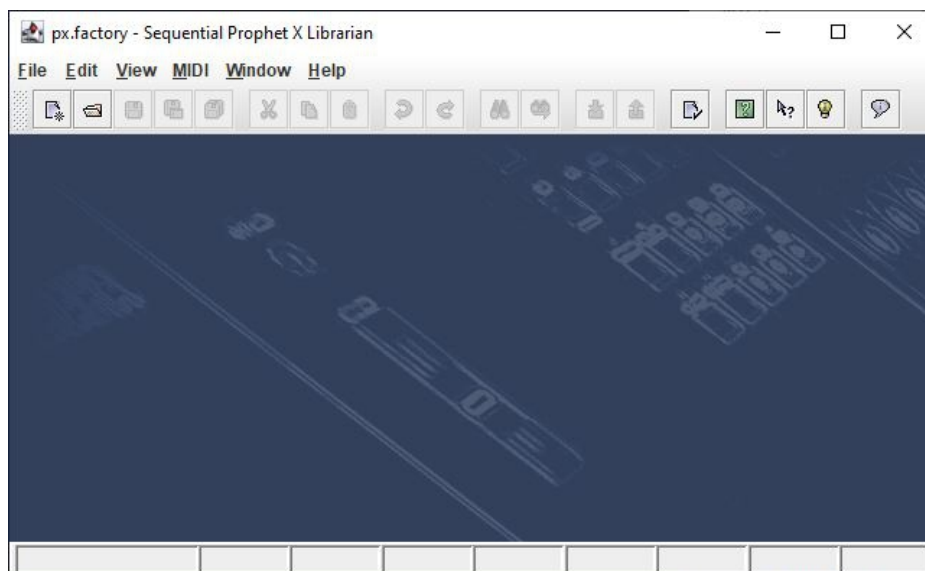
```
java -jar pxfactory.jar
```


The px.factory Window

The main px.factory window on OS X looks like this



The main px.factory window on Windows looks like this



The px.factory user interface is a Multiple Document Interface (MDI), which allows you to have more than one file open at a time.

The interface is divided into the now familiar layout of (from top to bottom):

- The Title Bar
- The [Menu Bar](#) (on OS X this appears at the top of the screen)
- The [Tool Bar](#)
- The [Desktop](#)
- The [Status Bar](#)

Menu Bar

The kronos.factory menu bar provides a means of selecting common operations; typically those that affect a complete file, or those that don't fit naturally upon a context sensitive popup menu.

[Keyboard Shortcuts](#) are provided for common operations.

kronos.factory also provides context sensitive popup menus, which are accessible by right clicking over data shown in a [Child Window](#). Naturally the contents of the popup menus varies according to the type of data over which the menu was popped up.

Main Menu Headings

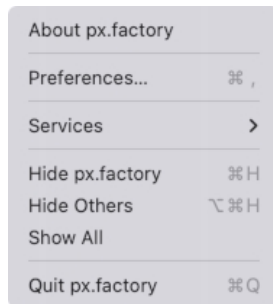
The following menu headings are available.



- [Apple Application Menu](#) (OS X only)
- [File Menu](#)
- [Edit Menu](#)
- [View Menu](#)
- [MIDI Menu](#)
- [Window Menu](#)
- [Help Menu](#)

The Application Menu

The **Application** menu is available on OS X to provide a native look and feel for OS X users; some menu options from the standard menus are migrated here (where OS X Users would expect them to be).



About px.factory...

This menu provides the standard OS X application menu, with the following px.factory specific options along with the standard OS X options for Services (none defined for px.factory) and Window control.

Displays the px.factory [About Dialog](#), which provides some basic information about the program, such as the version number.

Preferences ...

Opens the px.factory [Preferences Dialog](#).

Quit px.factory

Exits the px.factory application.

If any files have been modified and have not been saved, you will be given the option of saving them.

The File Menu

The **File** menu provides the following options (OS X and Windows versions are shown):

New ...	⌘N	New ...	Ctrl-N
Open ...	⌘O	Open ...	Ctrl-O
Close	⌘W	Close	Ctrl-W
Close All	⇧⌘W	Close All	Ctrl+Shift-W
Save	⌘S	Save	Ctrl-S
Save As ...	⇧⌘S	Save As ...	Ctrl+Shift-S
Save All ...	⌘L	Save All ...	Ctrl-L
Open Work Space...	⌘O	Open Work Space...	Ctrl+Alt-O
Save Work Space ...	⌘S	Save Work Space ...	Ctrl+Alt-S
Export to		Export to	
1: /Development/Prophet/Replicants-User1-Sample1.syx		Exit	
2: /Development/Prophet/Prophet X Program Bank U1.syx		1: E:\Prophet\Replicants-User1-Sample1.syx	
3: /De.../Prophet X Program Banks U1-U4, Play List.syx		2: E:\Prophet\Prophet X Program Bank U1.syx	
4: /Development/Prophet/pxFactory_1.syx			
5: /Development/Prophet/Test 3.syx			
6: /De.../Prophet/Prophet X Global Parameters.syx			

New ...

Shows the [Create New File](#) Dialog, which allows you to create a new file.

Open ...

Shows the [Open File](#) Dialog, which allows you to open a specified file.

Save

Saves a file using its current name.

This option is only enabled if your copy of px.factory is [registered](#).

If the file in question is a newly created file, then the [Save File](#) Dialog is brought up to allow a name to be entered.

Save As ...

Saves the selected file after a name has been entered in the [Save File](#) Dialog.

This option is only enabled if your copy of px.factory is [registered](#).

Save All

Saves all files that have been modified using their current names.

This option is only enabled if your copy of px.factory is [registered](#).

Open Workspace ...

This option allows you to select a previously saved [Workspace](#), which is a Collection of open windows, and their size and position within the main px.factory window.

When you select this option, a dialog opens which allows you to select an px.factory workspace file (EXW).

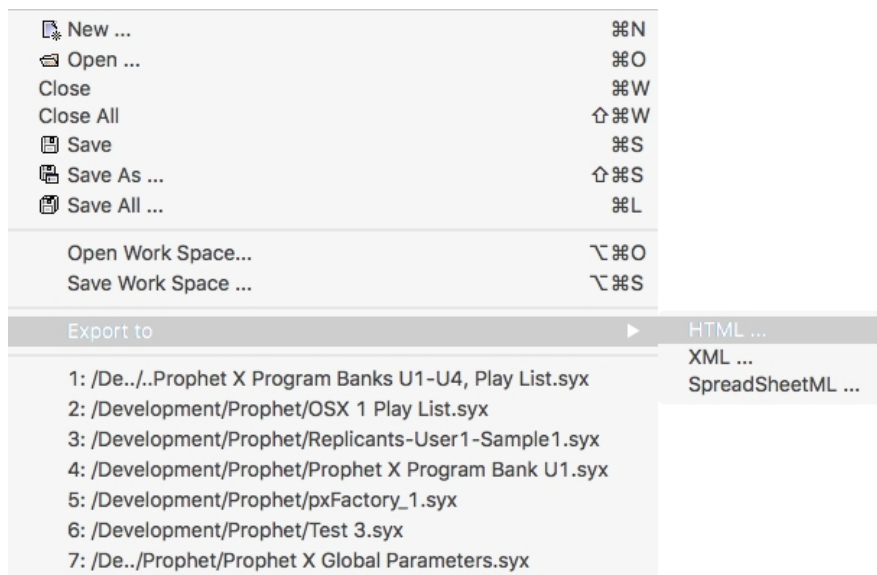
Save Workspace ...

This option allows you to save the current size and position of the main px.factory window, and the arrangement of the open windows within px.factory to a [Workspace](#) file.

This option is only enabled if your copy of px.factory is [registered](#).

Export to

This option brings up the following sub menu:



These options allow you to [Export](#) the data in the currently selected file to a variety of different formats.

Exit

This menu option is not available if the host operating system is OS X. On OSX there is a **Quit** menu option in the [px.factory Application Menu](#).

Exits the px.factory application.








If any files have been modified and have not been saved, you will be given the option of saving them.

Recently Opened Files

The file menu also contains a list of recently opened files to allow easy access to files that you are currently working upon. The recently open files list appears at the bottom of the file menu.

The Edit Menu

The **Edit** menu provides the following options:

 Undo initialise Program	⌘Z
 Redo initialise Program	⌘Y
Clear Undo Buffer	
<hr/>	
 Cut	⌘X
 Copy	⌘C
 Paste	⌘V
<hr/>	
 Find ...	⌘F
 Find Next	F3
Goto ...	⌘G

Undo

Undoes the previous edit stored in the [Undo Buffer](#).

Redo

Redoes the next edit stored in the [Undo Buffer](#).

Clear Undo Buffer

Clears the [Undo Buffer](#).

Cut

Places a copy of the selected Object(s) onto the [Clipboard](#) and deletes the selected Object(s) from the file.

Copy

Places a copy of the selected Object(s) onto the [Clipboard](#).

Paste

Copies the Object(s) on the [Clipboard](#) into the highlighted location within the current file. If you cannot perform a paste at this location (e.g. trying to copy a Set List onto a Program) then this menu item is disabled.

Find ...

px.factory supports the ability to find Objects by name.

Searches start from the currently selected Object.

Selecting the **Find ...** option brings up the [Find](#) dialog, that allows you to specify the string to search for, along with options that can be used to refine the search.

Find Next

If you have defined the **Find** criteria within the **Find** dialog, then this option will repeat the search using the same criteria.

Goto ...

Selecting this option brings up the [Goto](#) Dialog, which allows you to quickly navigate to a numbered Object.

The View Menu

The **View** menu provides the following options:

<input checked="" type="checkbox"/> Tool Bar	<input checked="" type="checkbox"/> Tool Bar	<input checked="" type="checkbox"/> Tool Bar
<input checked="" type="checkbox"/> Status Bar	<input checked="" type="checkbox"/> Status Bar	<input checked="" type="checkbox"/> Status Bar
<input checked="" type="checkbox"/> No Split	No Split	No Split
Horizontal Split	<input checked="" type="checkbox"/> Horizontal Split	Horizontal Split
Vertical Split	Vertical Split	<input checked="" type="checkbox"/> Vertical Split
<input checked="" type="checkbox"/> Tree	Left Pane ▶	Top Pane ▶
Table	Right Pane ▶	Bottom Pane ▶
Refresh	Refresh	Refresh

Note how some of the options change depending on whether or not the [Child Window](#) is split.

The **View** menu provides the following options upon PC:

<input checked="" type="checkbox"/> Tool Bar	<input checked="" type="checkbox"/> Tool Bar	<input checked="" type="checkbox"/> Tool Bar
<input checked="" type="checkbox"/> Status Bar	<input checked="" type="checkbox"/> Status Bar	<input checked="" type="checkbox"/> Status Bar
<input checked="" type="radio"/> No Split	<input type="radio"/> No Split	<input type="radio"/> No Split
<input type="radio"/> Horizontal Split	<input checked="" type="radio"/> Horizontal Split	<input type="radio"/> Horizontal Split
<input type="radio"/> Vertical Split	<input type="radio"/> Vertical Split	<input checked="" type="radio"/> Vertical Split
<input checked="" type="radio"/> Tree	Left Pane ▶	Top Pane ▶
<input type="radio"/> Table	Right Pane ▶	Bottom Pane ▶
Refresh	Refresh	Refresh
Preferences ...	Preferences ...	Preferences ...

Note how some of the options change depending on whether or not the [Child Window](#) is split.

When the [Child Window](#) is split the **Left/Top Pane** and **Right/Bottom Pane** menu options have the following sub menu.

<input checked="" type="checkbox"/> Tool Bar	
<input checked="" type="checkbox"/> Status Bar	
No Split	
<input checked="" type="checkbox"/> Horizontal Split	
Vertical Split	
Left Pane ▶	<input checked="" type="checkbox"/> Tree
Right Pane ▶	Table
Refresh	

Tool Bar

When checked the px.factory [Tool Bar](#) is visible.

Status Bar

When checked the px.factory [Status Bar](#) is visible.

No Split

When checked the current [Child Window](#) is not split.

Horizontal Split

When checked the current [Child Window](#) is split horizontally.

Vertical Split

When checked the current [Child Window](#) is split vertically.

Tree

When checked the main/left/top pane in a [Child Window](#) displays data in a [Tree View](#).

Table

When checked the main/right/bottom pane in a [Child Window](#) displays data in a [Table View](#).

Refresh

Refreshes all open [Child Windows](#).

This command is provided in case the screen is not properly updated following an edit operation. There are a couple of odd bugs I cannot track down on screen updates. So if you come across them as well, then this command will force a redraw of the windows.

Preferences ...

This menu option is not available if the host operating system is OS X. On OSX there is a **Preferences...** menu option in the [px.factory Application Menu](#).

Opens the px.factory [Preferences Dialog](#).

The Window Menu

The **Window** menu provides the following options:

Tile Horizontal	
Tile Vertical	⌘ V
Maximise Windows	⌘ M
Restore Windows	⌘ R
Minimise Windows	⌘ I
✓ View Prophet X Synthesizer Data	
Px Synthesizer	
Replicants-User1-Sample1.syx	
Prophet X Program Banks U1-U4, Play List.syx	

In the example given above, you'll see that after the predefined menu options there are menu options that allow you to select the windows currently open within px.factory.

Tile Horizontal

Arranges all non-minimised [Child Windows](#) along the horizontal axis.

Tile Vertical

Arranges all non-minimised [Child Windows](#) along the vertical axis.

Maximise Windows

Maximises all [Child Windows](#) so that they all occupy the size defined by the desktop.

Restore Windows

Restores all maximised/minimised [Child Windows](#) to their normal state.

Minimise Windows

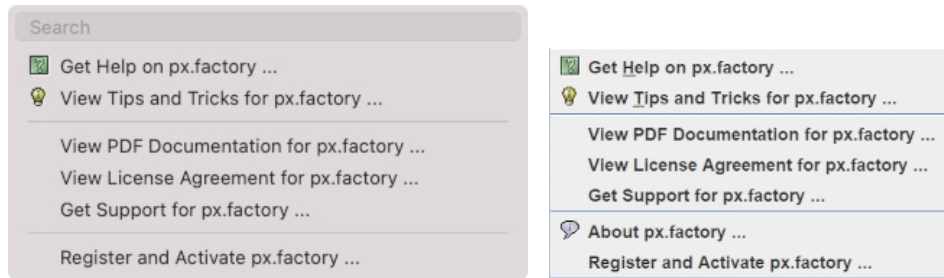
Reduces all [Child Windows](#) to their iconised state.

View Prophet X Synthesizer Data

Opens or closes the [Prophet X Synthesizer Window](#) which provides a view on preset and user DX synth data.

The Help Menu

The **Help** menu provides the following options (OS X and Windows versions are shown):



Search

OS X only. This is a standard OS X Menu function currently not working with px.factory. One day....

Get Help on px.factory ...

Displays the px.factory [Online Help](#) system.

View Tips and Tricks for px.factory ...

Displays the px.factory [Tips and Tricks](#) Dialog.

View PDF Documentation for px.factory ...

Displays the PDF documentation for px.factory. Note that there needs to be a registered viewer for PDF files on your system for this command to work

View License Agreement for px.factory ...

Displays the px.factory [License Agreement](#) in a dialog box.

View License Agreement for px.factory ...

Provides a link to the [x.Librarian's Support Page](#).

About px.factory ...

This menu option is not available if the host operating system is OS X. On OSX there is an **About px.factory** menu option in the [px.factory Application Menu](#).

Displays the px.factory [About Dialog](#), which provides some basic information about the program, such as the version number.




















Register and Activate px.factory ...

Displays the px.factory registration and activation Dialog.

See the [Registration Section](#) for details of how to register px.factory.

Tool Bar

The Tool Bar contains buttons for common operations, such as opening and saving files.

-  Displays the [New File](#) Dialog to allow you to select the type of file to create
-  Displays the [Open File](#) Dialog to allow you to select a file to open
-  [Saves](#) the currently selected file
-  Displays the [Save As](#) Dialog to allow you to select a different name for the currently selected file
-  [Saves](#) all the open files to disk
-  Cuts the selected Objects and places them on the [Clipboard](#)
-  Copies the selected Objects and places them on the [Clipboard](#)
-  Pastes the contents of the [Clipboard](#) to the current selection. If you cannot perform a paste at this location (e.g. trying to copy a Set List onto a Program) then this menu item is disabled
-  [Undos](#) the previous edit
-  [Redos](#) the previous edit
-  Displays the [Find](#) Dialog to allow you to define the search criteria and then search for the first occurrence
-  Searches for the next occurrence of the search criteria defined in the [Find](#) Dialog
-  Imports Programs/Playlist/Global Parameters from the Prophet X via the [MIDI](#) interface
-  Exports Programs/Playlist/Global Parameters to the Prophet X via the [MIDI](#) interface
-  Displays the [Preferences](#) dialog where you can configure px.factory
-  Displays the px.factory [Online Help](#) index page
-  Enables [Context Sensitive](#) Help
-  Displays the [Tips and Tricks](#) Dialog
-  Displays the [About](#) Dialog

The Save buttons are only enabled if px.factory is [Registered](#).

The Tool Bar can be dragged and made a floating toolbar by clicking and dragging the handle to the left of the Tool Bar.

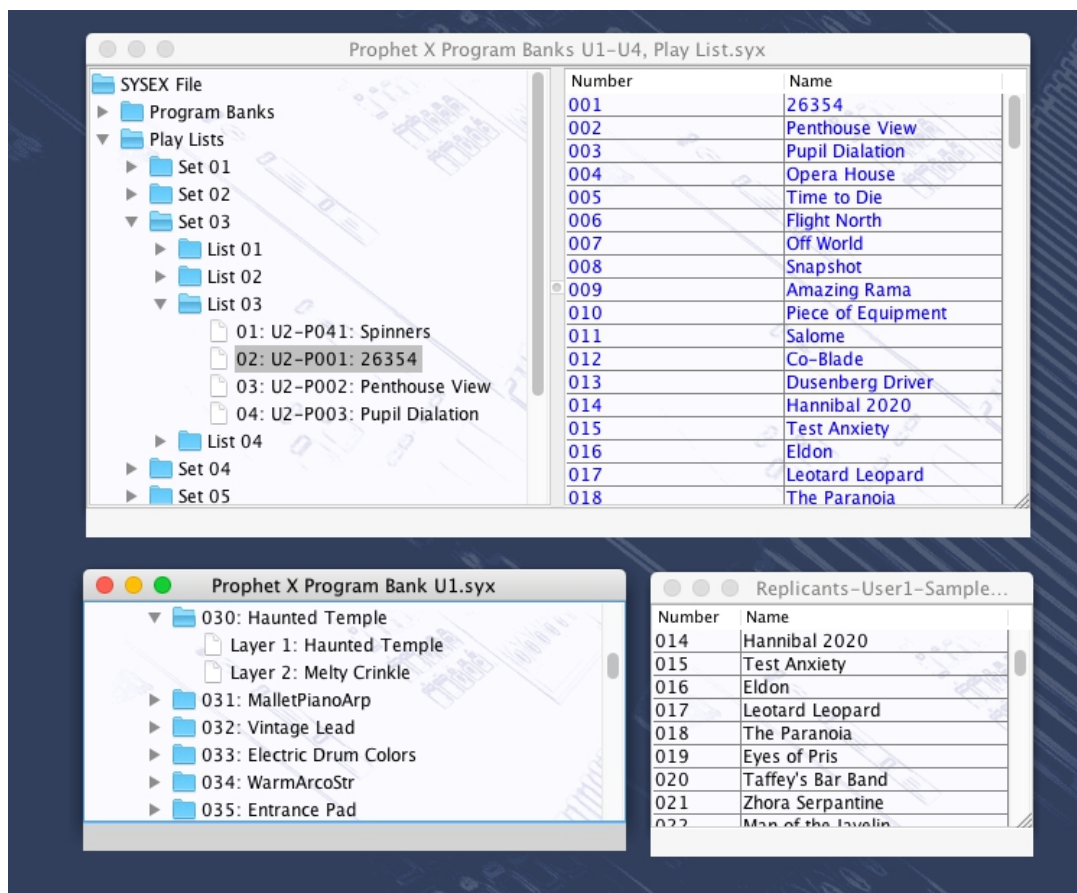
The Tool Bar can also be dragged to a different window border other than the top border.

The Tool Bar can be hidden/unhidden by the **Tool Bar** menu option on the [View](#) menu.

The Tool Bar whilst floating can be docked once more by closing it.

Desktop

The Desktop portion of the window is where the Prophet X data is portrayed. Several files may be opened and shown within [Child Windows](#) upon the Desktop portion of the window. The following example shows three [Child Windows](#) open on the Desktop.



The [Window Menu](#) provides menu options to:

- Tile (either vertically or horizontally) all [Child Windows](#) that are not iconised
- Minimise all [Child Windows](#)
- Maximise all [Child Windows](#)
- Restore all [Child Windows](#)
- Select a specific [Child Window](#)

When a file is opened, a [Child Window](#) is added to the Desktop Area.

Child Window

The Child Window provides the **View** upon the data within the file it is associated with.

The Child Window is maximised by default when it is opened, and can be minimised, restored or closed by the buttons to the right of the Child Window's title bar.

The menu options on the [View](#) menu can be used to change the View between a [Tree](#) view and a [Table](#) view, and a Child Window can be either split horizontally or vertically split to show any of the following combinations:

- A [Tree](#) and a [Table](#)
- A [Table](#) and a [Tree](#)
- Two [Trees](#)
- Two [Tables](#)

The default Child Window View applied when a file is opened is set in the [Default View Tab](#) of the [Preferences](#) Dialog.

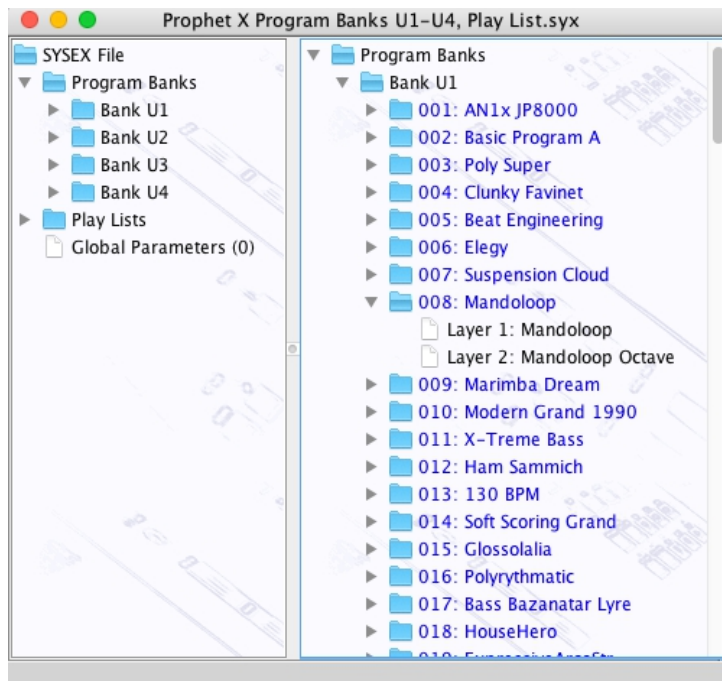
The following example shows a Child Window that is split and showing both a [Tree View](#) and a [Table View](#) on the same data.



The [Tree](#) and [Table](#) views support drag [move](#) and drag [copy](#) operations, and the views will automatically scroll whilst drag operations are in progress.

Tree View

The **Tree View** shows the file data as a hierarchical tree, and can show more information than the [Table View](#). The following example shows a split [Child Window](#) with two Tree Views, one collapsed showing the top level **Collections**, including all the available Program Banks, and one showing a Program (U-1 008) that has been expanded to show more information on the Program.



Basically, information is shown as a set of hierarchical **Objects**. Objects may have children attached to them, and they themselves may have further child Objects (e.g. a Combi Object, has a child Object for all of its Timbres, and this Object groups all the Timbres).

The basic tree structure is a **Root Object** at the top of the tree under which are various Object **Collections** which themselves contain further data. The top level Collections that can be found under the root node varies upon the file contents

Objects are displayed in one of three colours:

- Black if the Object is not referenced by another Object
- Blue if an Object is referenced by another Object (e.g. a Voice being used within a Performance, or a Sample being used by a Wave)
- Red if the Object is "empty" (e.g. a Program Collection)

The Tree View allows multiple Objects to be selected, which is useful for performing operations only upon certain Objects. You can only perform operations upon Objects of the same type.

If you wish to perform an operation upon all Objects in a Collection, simply select the Collection containing the Objects.

If you right click over an object or selection of objects within the Tree View, a context sensitive popup menu will appear that provides the editing operations that can be performed upon the object(s).

Objects can be [Copied and Moved](#) either within or between Child Window Views using **Drag and Drop**.

Table View

The Table View provides a more high level overview of the data than the [Tree View](#), and the representation of the data is more compact. The following example shows a [Child Window](#) with a single Table View providing information on a file's Programs in one of the Program Banks.



Number	Name
001	AN1x JP8000
002	Basic Program A
003	Poly Super
004	Clunky Favinet
005	Beat Engineering
006	Elegy
007	Suspension Cloud
008	Mandoloop
009	Marimba Dream
010	Modern Grand 1990
011	X-Treme Bass
012	Ham Sammich
013	130 BPM
014	Soft Scoring Grand
015	Glossolalia
016	Polyrhythmic
017	Bass Bazanatar Lyre
018	HouseHero
019	ExpressiveArcoStr
020	Everypluck
021	Warm Harmonic MW
022	Strong Pound
023	Thy Saviour cometh
024	GrimyDrumz

The Table View can only show one Collection of Objects at a time.

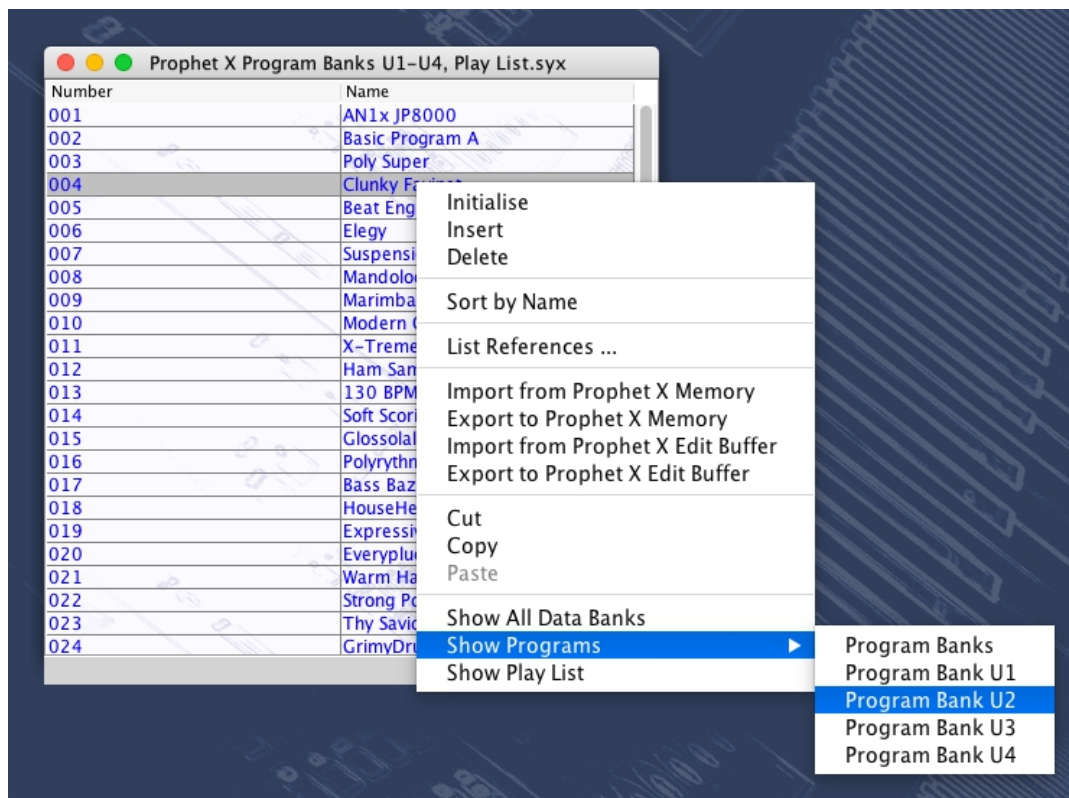
Each row within the table shows one Object.

Objects can be [Copied and Moved](#) either within or between Child Window Views using **Drag and Drop**.

If you right click over an object or selection of objects within the Table View, a context sensitive popup menu will appear that provides the editing operations that can be performed upon the object(s). The cells in a Row that contain the Object **Category** and Object **Name** (if available) can be directly edited by clicking in them.

The context menu also provides the options to select what data view. In the example below, the table is currently showing Program Bank U1, but a selection is being made via the menu to change to the U2 Bank.

You can also double click a high level collection (e.g. Program Bank Collection, Program Banks) and px.factory will move down to the next level until you get to discrete objects (e.g. Program). You can double click with the ALT key (OPTION key on OS X) held down to traverse upwards.



Status Bar

/Development/Prophet/pxFact...	SYSEX File					Program
--------------------------------	------------	--	--	--	--	---------

The Status Bar provides the following pieces of information (from left to right) for the [Child Window](#) which is currently selected:

- The name of the file currently open, if any. If no file has been selected then this field is blank
- The type of the file
- Whether or not the file has been modified
- If the left/top split pane is a table this indicator identifies what the table is showing (Programs, Play List, etc.)
- If the right/bottom split pane is a table this indicator identifies what the table is showing (Programs, Play List, etc.)

px.factory Program Features

kronos.factory provides the following features:


- [File Operations](#)
- [Undo/Redo Support](#)
- [Clipboard Support](#)
- [Find and Goto Support](#)
- [Editing Features](#)

File Operations

px.factory supports the following file operations:

- [Create New Files](#)
- [Open Existing Files](#)
- [Save Files](#)
- [Open and Save Workspaces](#)
- [Export Data](#)

Create New File

When the **New File ...** option from the [File](#) Menu or the  button on the [Toolbar](#) is selected, the following dialog is shown:



This dialog allows you to select what banks are created in the file from the following:


- Program Bank U1
- Program Bank U2
- Program Bank U3
- Program Bank U4
- Performance Play List
- Global Parameters

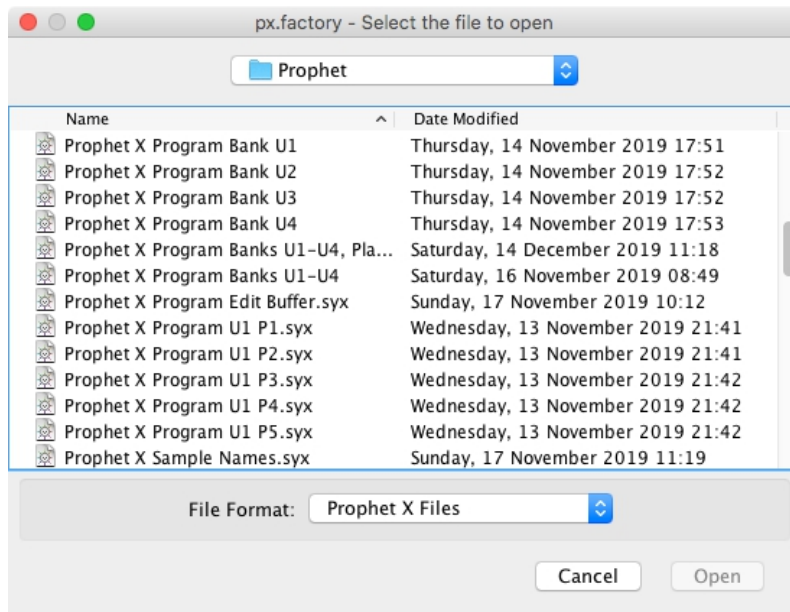
The initial values for these settings are setup in the [New File Default Preferences](#) Tab in the [Preferences](#) Dialog.

The new data file is given a name in the form of "pxFactory_n", where **n** is a number that increments every time a file is created.

Note, this is different from initialising data in a file, as a totally new file is created.

Open Existing File

When the **Open ...** option from the [File](#) menu or the  button on the [Toolbar](#) is selected, the following dialog will appear allowing a file to be selected.



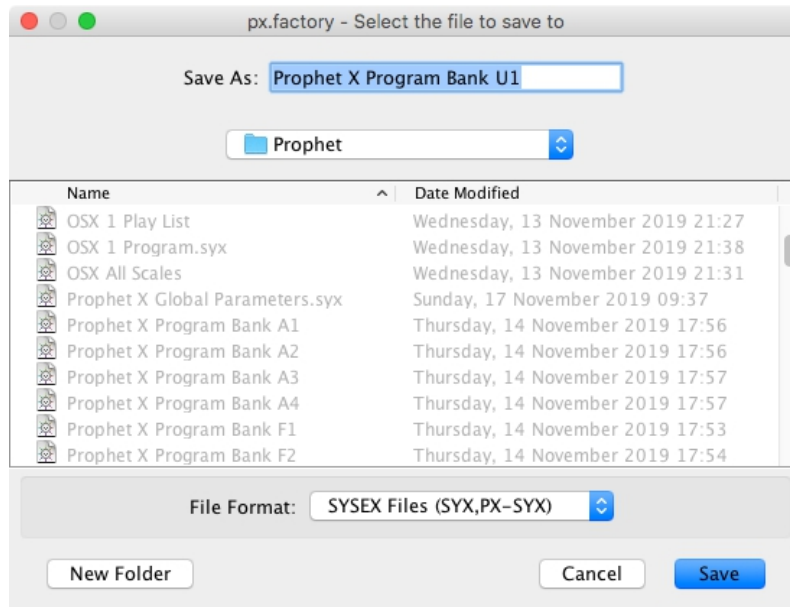
px.factory checks to ensure that the selected file contains valid Prophet X formatted data, and an error dialog will appear if the file is not valid.

If a file selected for opening is already open, px.factory will ask if you want to replace the file that is in memory.

Save Open Files

The saving of files is only available if px.factory is [registered](#).

When you save a file to disk using: the **Save As ...** command from the [File](#) Menu, or when a file created using the **New ...** command from the [File](#) Menu is saved for the first time, the following dialog appears to allow you to select a name for the file.



Workspaces

px.factory allows you to save the current size and position of the main px.factory window, and the arrangement of the open [Child Windows](#) within px.factory to a file known as a **Workspace**.

This allows you to save a **Workspace** you are using for future recall at a later date.

The following parameters for each [Child Window](#) are remembered:

- Window size, position and status (normal, maximised or minimised)
- Horizontal/Vertical Splits, including split position
- Pane type(s) (e.g. Tree, Voice Table or Performance Table)
- Tree node expansion status
- Table column widths
- Scroll position of the pane(s)
- Selection state of the tree nodes, or table rows

Export Data

px.factory allows you to export the data in the currently selected file to a variety of different formats.

- [HTML](#)
- [XML](#)
- [SpreadSheetML](#)

HTML ...

This option allows you to export a listing of the data in HTML format.

XML ...

This option allows you to export a listing of the data in XML format.

This export option is intended to provide a means of importing Prophet X data into other programmes, e.g. a database

SpreadSheetML ...

This option allows you to export a listing of the data in Microsoft SpreadSheetML format.

This export option is intended to provide a means of importing Prophet X data into either Microsoft Word or Excel, which is a great way to start documenting your program collections.

Undo Buffer

px.factory supports a multi-level Undo/Redo Buffer facility on all edits.

The number of edits which can be stored is 100.

The Undo/Redo commands can be found both within the [Edit Menu](#), and on the [Toolbar](#).

Please note that the undo buffer is global across all open files, and is cleared when any file is closed. This may be improved in later releases.

The undo buffer can be cleared by the selecting the **Clear Undo Buffer** option on the [Edit Menu](#). The intention of this feature is to help cope with low memory situations, as it will free up the memory used by stored edits.

Clipboard Support

The clipboard works just like the clipboard in other applications. It allows you to place an Object or a copy of an Object upon the clipboard and then paste that Object elsewhere within a file or another file.

You can Cut/Copy either a single Object or a Collection of Objects onto the clipboard.

You can only paste from the clipboard to the current location in a file if it makes sense to do so. For example:

- You cannot paste a Play List onto a Program
- You cannot paste ten Programs at the end of a Program Collection if the selection is at Program 118 or higher

The clipboard is "local" to the application, and only Objects can be placed upon it.

px.factory does not support the system clipboard so it is not possible to cut and paste between and other applications. I don't think this is a major drawback because I cannot think of any circumstances where you would wish to do this!

Find and Goto Facilities

Find Dialog

kronos.factory supports the ability to find Objects either by name or by category (where an Object supports categories, such as Voices).

Searches start from the currently selected Object.

Selecting the **Find ...** option on the [Edit Menu](#) brings up the **Find** dialog, that allows you to specify the string to search for. In addition there are several options that can be selected



Find What	The text string to search for.
Match Case	If this option is checked then the search is case sensitive. If this option is not checked then the search is case insensitive.
Match Whole Name	If this option is checked then the search string must match the entire name of an Object. E.g. "pan" would match a Voice called "pan", but not a Voice called "pandora". If this option is not checked then the search can match a sub string within a name. E.g. "dor" would match a Voice named "dor" and a Voice called "pandora".
Use Wild Cards	If checked kronos.factory will treat the "Find What" text as a "regular expression" to allow you more control over the search, such as matching text at the start of a name, or the end of a name, words with numbers in, etc.

The regular expression language used for kronos.factory wild card support is the Perl Regular Expression Syntax. Rather than try and write up this powerful syntax myself into the kronos.factory documents, I'll direct you to the [Perl Regular Expression Syntax](#) page for a description of how to use this powerful feature.

Goto Dialog

Selecting this option on the on the [Edit Menu](#) brings up the **Goto** Dialog



Item Type	The type of Object to go to <ul style="list-style-type: none">• The list of types of Objects that can be selected depends upon the contents of the file.• If a Table View is selected then you can only go the type of objects shown by the Table
Item Number	The number of the Object to go to.

Editing Features

px.factory supports the following editing features:

- [General Editing Features](#)
- [Synth Editing Features](#)
- [Program Editing Features](#)
- [Play List Editing Features](#)
- [Global Parameters Editing Features](#)

General Editing Features

px.factory supports many editing features, some of which are classed as general features that are applicable to all or most Objects.

The general features are described here in one place for convenience:

- [Copy/Move Object](#)
- [Initialise Object](#)
- [Rename Object](#)
- [Insert Object](#)
- [Delete Object](#)
- [List Object References](#)
- [Sort Objects](#)
- [Export Objects](#)

Copy/Move Object

px.factory makes extensive use of drag and drop for copying and moving Objects.

You can drag copy or move Objects within a file or between files, and you can drag single items or multiple items.

The default drag operation is to move an Object, and there are two types of move operations:

- Moving an Object within its Parent Collection (e.g. moving a Program from location 1 to location 4)
- Moving an Object to a new Parent Collection (e.g. moving a Program from one file to another file)

When you move an Object to a new Parent Collection then the original Object is removed and replaced with an initialised Object of the same type.

To copy an Object using drag and drop, you must have the **DRAG MODIFIER** key held down during the operation. The **DRAG MODIFIER** key is platform dependent. On the PC platform it is the **CTRL** key.

You can tell that a copy operation is in progress as the drag image icon has a + sign visible.

When you drag multiple items they must be of the same type. For example you can drag several Programs, but not Programs and Play Lists together.

px.factory provides the following drag/drop feedback to assist in determining the effects of a drag/drop operation:

- Cursor: Indicates if the move is either a copy or a move, or if the drop is invalid (e.g. trying to drag a Program onto a Play List)
- Drag image: A simple pictorial representation of what is being dragged
- Drop Point: px.factory shows the drop point as either a cue line drawn between Objects if an Object is being moved within its parent (e.g. a Program within a Program bank), or a cue rectangle drawn around the drop location for all other moves and copy operations which are valid

When a [Child Window](#) is split, Objects can be dragged and dropped between either side of the split point. This is handy, for example, for dragging Objects between the start of a Collection and the end of a Collection.

The [Tree](#) and [Table](#) views will auto-scroll in drag and drop operations, when you move the cursor to either the top or the bottom of the pane and hold it there.

If you drag an Object onto a [Tree View](#) and over a valid parent Object (e.g. a Program over a Program Collection) and the parent Object is collapsed within the Tree, and you hold the cursor over the Parent Object for a few seconds it will automatically expand.

px.factory will maintain the references to Objects in a Collection when moving Objects within a file, as follows:

Collection Type	Objects Updated
Program	Play List - Set - Set List - Program Slots

For example if you move a Program within a file's Program Bank from Location 2 to Location 10, then any Play Lists which are referencing Programs at Locations 2 and above will be updated to ensure they still reference the same Programs after the move operation as they were before the move operation.

The ability to update references following a move or to copy referenced Objects during a move to a new file can be selectively turned on or off within the [Preferences](#) Dialog.

Initialise Object

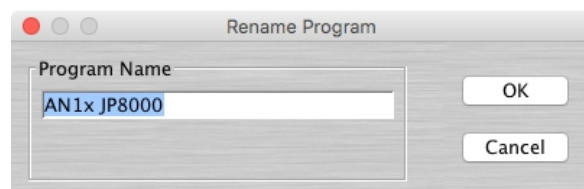
Initialises the selected Object to its default state.

Multiple Objects of the same type can be selected for initialisation.

The **Initialise** command is found upon the **Context Sensitive Menu** in both the [Tree View](#) and the [Table View](#).

Rename Object

Renames the selected Object using the following dialog:



Only one Object can be selected for renaming.

The **Rename** command is found upon the **Context Sensitive Menu** in the [Tree View](#)

To rename Objects in the [Table View](#), simply click in **Category Cell** of the required Object to select the Category value from a drop down list, or double click in the **Name Cell** of the required Object to edit the name.

Insert Object

Inserts an Object into a Collection at the current location.

The **Insert** command is found upon the **Context Sensitive Menu** in both the [Tree View](#) and the [Table View](#).

You can insert more than one Object at a time by selecting multiple Objects prior to executing the **Insert** command.

For example if you wish to insert three Combis at location 10, select Performances 10, 11 and 12 and then execute the **Insert** command.

Note that when you insert one or more Objects, then the commensurate number of Objects at the end of the Parent Collection are removed if required. This is necessary to maintain the correct number of Objects in a Collection.

For example, a Combi Branch Collection in a PCG file can only contain 128 Combis, so if two Combis are inserted into the Collection then two Combis must be removed from the end of the Collection to maintain the correct size.

px.factory will maintain the references to Objects in a Collection when inserting Objects, as follows:

Collection Type	Objects Updated
Program	Play List - Set - Set List - Program Slots

For example if you insert a Program into a file Program Bank at Location 10, then any Play Lists which are referencing Programs in the bank at Locations 10 and above will be updated to ensure they still reference the same Programs after the insert operation as they were before the insert operation.

Delete Object

Deletes an Object at the current location.

The **Delete** command is found upon the **Context Sensitive Menu** in both the [Tree View](#) and the [Table View](#).

You can delete more than one Object at a time by selecting multiple Objects prior to executing the **Delete** command.

For example if you wish to delete three Combis at location 10, select Performances 10, 11 and 12 and then execute the **Delete** command.

Note that when you delete one or more Objects, then a commensurate number of Objects at the end of the Parent Collection are added if required. This is necessary to maintain the correct number of Objects in a Collection.

For example, a Program Bank Collection in a PCG file must contain 128 Voices, so if two Programs are deleted from the Collection then two Programs must be added to the end of the Collection to maintain the correct size.

px.factory will maintain the references to Objects in a Collection when deleting Objects, as follows:

Collection Type	Objects Updated
Program	Play List - Set - Set List - Program Slots

For example if you delete a Program from Program Bank at Location 10, then any Play Lists which are referencing Programs in that Bank at Locations 10 and above will be updated to ensure they still reference the same Programs after the delete operation as they were before the delete operation.

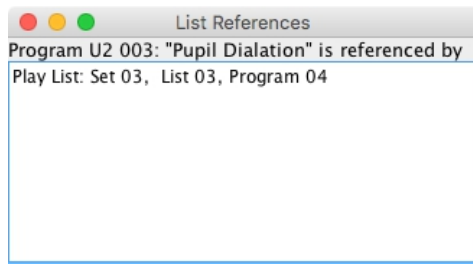
List Object References

The **List References** command is found upon the **Context Sensitive Menu** in both the [Tree View](#) and the [Table View](#).

The **List References** command is only available if:

- Only one Object is selected
- The Object selected is referenced by another Object, which is indicated by the Object being coloured [blue](#)

The references are shown as follows:



The Object being referenced is shown at the top of the Dialog followed by the Objects referencing it.

Sort Objects

The **Sort by Name** command is found upon the **Context Sensitive Menu** in both the [Tree View](#) and the [Table View](#).

In the [Tree View](#) the sort options are accessed via the **Context Sensitive Menu** for the Collection that you wish to sort (e.g. a Program Bank Collection to sort Programs).

In the [Table View](#) the sort options are accessed via the **Context Sensitive Menu** for any Object. The sort can also be initiated by clicking in the **Name** column header.

Synth Editing Features

px.factory supports the following editing operations upon the **Synth Object** within the [Tree View](#) of a [Child Window](#):

- [Move](#)
- [Copy](#)
- [Initialise](#)
- Add - If not all main collections are present in the file (e.g. Program Banks, Play List, etc.) then this menu will be present allowing you to add in the items that are not present
- [Import From Prophet X Memory](#)
- [Export To Prophet X Memory](#) (only available if px.factory is [registered](#))

Import from Prophet X Memory

Imports Programs, Play Lists and Global Parameters from the Prophet X Memory locations into the currently selected [Child Window](#).

If a data object does not existing in the file then it is not imported. Play Lists and Global Parameters will be ignored unless the relevant options in the [MIDI Preferences](#) dialog are checked.

Export to Prophet X Memory

Exports Programs, Play Lists and Global Parameters from the currently selected [Child Window](#) to the Prophet X Memory locations.

If a data object does not existing in the file then it is not exported. Play Lists and Global Parameters will be ignored unless the relevant options in the [MIDI Preferences](#) dialog are checked.

To access these features, right click over the **Synth Object** to activate the **Synth Object's** Context Sensitive Menu.

The **Synth Object** is only shown within a [Tree View](#) and is the first Object in the [Tree View](#).

Program Editing Features

Program Bank Collection Editing Features

px.factory supports the following editing operations upon the **Program Bank Collection** shown within the [Tree View](#) or [Table View](#) of a [Child Window](#):

- [Copy/Move](#)
- [Initialise](#)
- [Delete](#)
- Add Program Bank - if there are unused banks within the file, then the unused bank names are presented as sub menu items for selection to create a new bank
- [Import From Prophet X Memory](#)
- [Export To Prophet X Memory](#) (only available if px.factory is [registered](#))
- [Import from Prophet X Edit Buffer](#)
- [Export to Prophet X Edit Buffer](#) (only if available px.factory is [registered](#))
- [Cut to Clipboard](#)
- [Copy to Clipboard](#)
- [Paste from Clipboard](#)

To access all features other than Move and Copy (which you do via "Drag and Drop"), right click over the **Program Collection** to activate the **Program Collection's** Context Sensitive Menu.

Program Bank Editing Features

px.factory supports the following editing operations upon the **Program Banks** shown within the [Tree View](#) or [Table View](#) of a [Child Window](#):

- [Copy/Move](#)
- [Initialise](#)
- Renumber - if there are unused banks within the file, then the unused bank numbers are presented as sub menu items for selection to allow this bank to be renumbered
- [Delete](#)
- [Sort by Name](#)
- [Import From Prophet X Memory](#)
- [Export To Prophet X Memory](#) (only available if px.factory is [registered](#))
- [Cut to Clipboard](#)
- [Copy to Clipboard](#)
- [Paste from Clipboard](#)

To access all features other than Move and Copy (which you do via "Drag and Drop"), right click over the **Program Collection** to activate the **Program Collection's** Context Sensitive Menu.

Program Editing Features

px.factory supports the following editing operations upon the **Program Objects** within the [Tree View](#) or [Table View](#) of a [Child Window](#):

- [Copy/Move](#)
- [Initialise](#)
- [Rename](#)
- [Insert](#)
- [Delete](#)
- [List References](#)
- [Import From Prophet X Memory](#)
- [Export To Prophet X Memory](#) (only available if px.factory is [registered](#))
- [Cut to Clipboard](#)
- [Copy to Clipboard](#)
- [Paste from Clipboard](#)

To access all features other than Move and Copy (which you do via "Drag and Drop"), right click over a **Program** or selection of **Programs** to activate the **Program's** Context Sensitive Menu.

In the [Tree View](#) you can drill down further into the structure of a Program. All objects will support:

- [Copy/Move](#)
- [Initialise](#)
- [Cut to Clipboard](#)
- [Copy to Clipboard](#)
- [Paste from Clipboard](#)

Program Layers can also be renamed. Layer A sets the name of the Program.

You can **DRAG COPY** a single Program onto a Play List Entry to set the reference to the Program.

Import from Prophet X Memory

Imports Programs from the Prophet X Memory locations into the currently selected [Child Window](#).

Export to Prophet X Memory

Exports Programs from the currently selected [Child Window](#) to the Prophet X Memory locations.

Import from Prophet X Edit Buffer

Imports the Program currently in the Prophet X Edit Buffer (i.e. the currently selected Program on the Prophet X) into the Program currently selected within the active [Child Window](#).

Note if you have more than one Program selected then this operation is not available

Export to Prophet X Edit Buffer

Exports the Program currently selected within the active [Child Window](#) to the Prophet XPropt X Edit Buffer.

Note if you have more than one Program selected then this operation is not available

Play List Editing Features

Play List Editing Features

px.factory supports the following editing operations upon the **Play List Object** shown within the [Tree View](#) or [Table View](#) of a [Child Window](#):

- [Copy/Move](#)
- [Initialise](#)
- [Delete](#)
- [Import From Prophet X Memory](#)
- [Export To Prophet X Memory](#) (only available if px.factory is [registered](#))
- [Cut to Clipboard](#)
- [Copy to Clipboard](#)
- [Paste from Clipboard](#)

To access all features other than Move and Copy (which you do via "Drag and Drop"), right click over the **Play List Collection** to activate the **Play List Collection's** Context Sensitive Menu.

Import from Prophet X Memory

Imports the Play List from the Prophet X Memory locations into the currently selected [Child Window](#).

Export to Prophet X Memory

Exports the Play List from the currently selected [Child Window](#) to the Prophet X.

Play List Set Editing Features

px.factory supports the following editing operations upon the **Play List Sets, Lists and Programs** shown within the [Tree View](#) or [Table View](#) of a [Child Window](#):

- [Copy/Move](#)
- [Initialise](#)
- [Insert](#)
- [Delete](#)
- [Cut to Clipboard](#)
- [Copy to Clipboard](#)
- [Paste from Clipboard](#)

To access all features other than Move and Copy (which you do via "Drag and Drop"), right click over the **Play List Sets, Lists and Programs** to activate the Context Sensitive Menu.

You can **DRAG COPY** a Programs onto a **Play List Set List Program** to set the references to them.

Global Editing Features

px.factory supports the following editing operations upon the **Global Parameters Object** shown within the [Tree View](#) or [Table View](#) of a [Child Window](#):

- [Copy/Move](#)
- [Initialise](#)
- [Delete](#)
- [Cut to Clipboard](#)

- [Copy to Clipboard](#)
- [Paste from Clipboard](#)

To access all features other than Move and Copy (which you do via "Drag and Drop"), right click over the **Global Parameters Object** to activate the **Global Parameters Object's** Context Sensitive Menu.

Preferences

V1.0 note: Some options listed here are for future program features, so may have no effect in V1.0

The px.factory preferences are accessed via the **Preferences ...** option on the [View](#) menu or the [px.factory Application Menu](#) on OS X, or the  button on the [Toolbar](#).

Selecting this option opens a Tabbed dialog with the following tabs which group related parameters:

- [New File Preferences](#)
- [Play List Preferences](#)
- [Check Preferences](#)
- [Default View Preferences](#)
- [MIDI Preferences](#)
- [Miscellaneous Preferences](#)

To view and modify a set of parameters, click on the relevant tab.

Note: The options to update Objects following an Object move within the same file (e.g. **Update Play List after Program Move**) also apply when Objects that can be referenced are inserted, deleted and sorted.

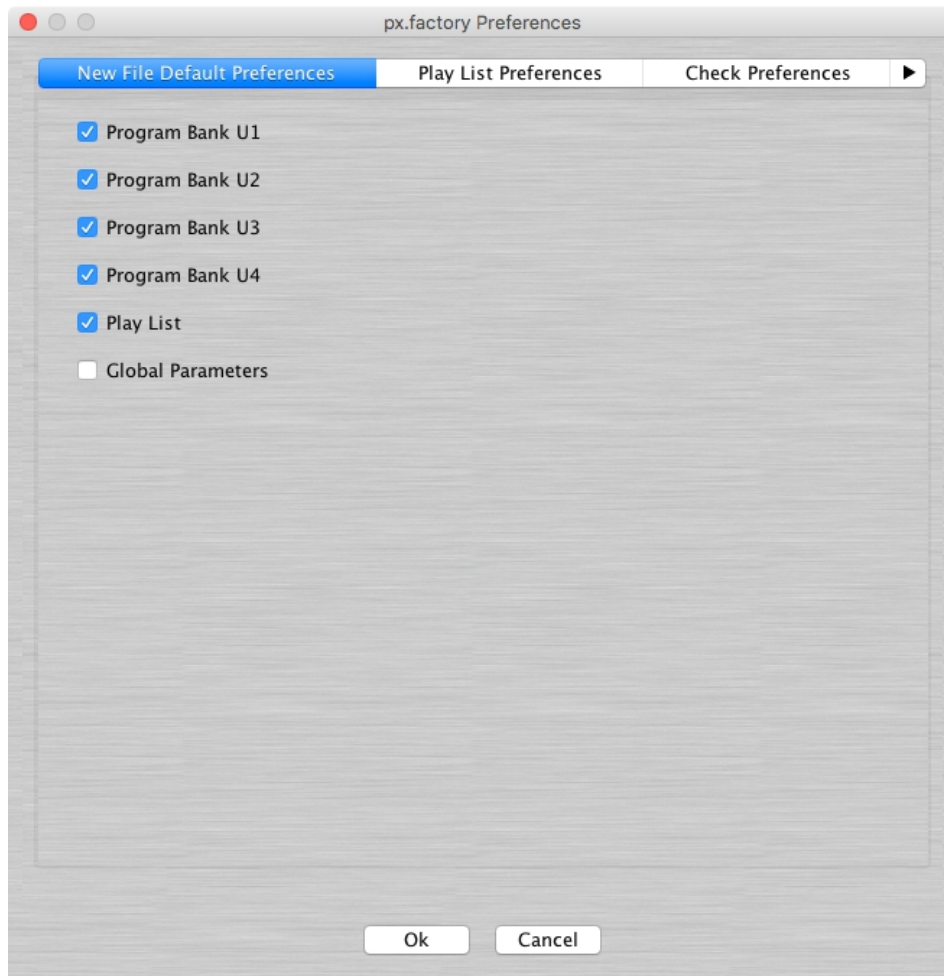
Note: The options to copy referenced Objects with "parent" Objects (e.g. moving Programs with a Play List) work by:

- Checking the destination file to see if the referenced Object exists
- If it does then the parent Object is updated after being moved to point to the existing Object in the destination file
- If not then px.factory checks to see if a blank or empty child Object is present in the destination file (e.g. in the case of moving a Program, a blank "Basic Program A" is looked for)
- If a blank Object is found then the child Object is copied and the parent Object is updated after being moved to point to the new Object in the destination file
- If no blank Object is found the px.factory gives up. Note this may result in a Partial edit (e.g. it may have only been possible to copy a subset of Objects due to the amount of free Objects in the destination file)
- The child Objects are only ever copied irrespective of whether the parent Object is being moved or copied. This is to avoid the problem that would occur when a child Object is moved (say a Program with Play List 1), where the source child Object (which would have been initialised after the move) is also referenced by other Parent Objects (e.g. another Play List entry)

All the above options are sensitive to what is being dragged. For example a Play List, Play List Set, Play List Set List, or Play List Set List Program can be dragged and the necessary Programs will be copied

New File Default Preferences

The New File Default Preferences tab is shown below:

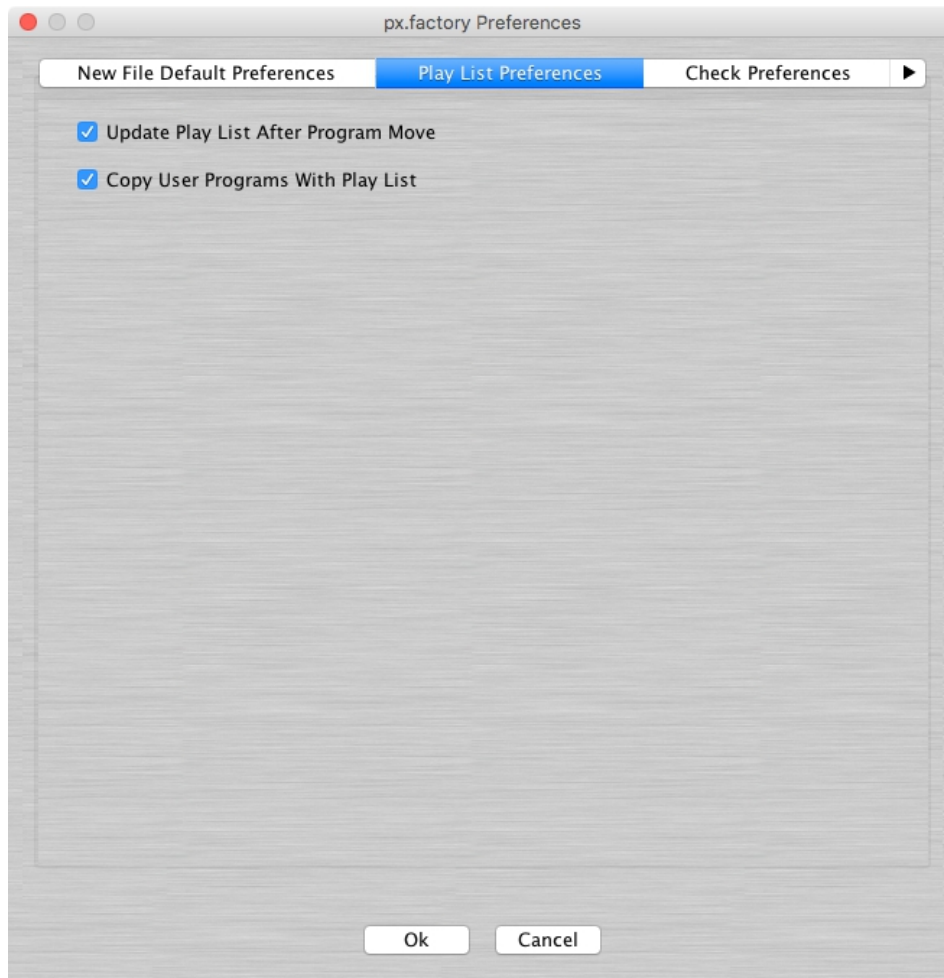


This tab has the following options:

Program Bank U1	When checked the Program Bank U1 checkbox in the New File Dialog will be initially checked when the Dialog is opened.
Program Bank U2	When checked the Program Bank U2 checkbox in the New File Dialog will be initially checked when the Dialog is opened.
Program Bank U3	When checked the Program Bank U3 checkbox in the New File Dialog will be initially checked when the Dialog is opened.
Program Bank U4	When checked the Program Bank U4 checkbox in the New File Dialog will be initially checked when the Dialog is opened.
Play List	When checked the Play List checkbox in the New File Dialog will be initially checked when the Dialog is opened.
Global Parameters	When checked the Global Parameters checkbox in the New File Dialog will be initially checked when the Dialog is opened.

Play List Preferences

The Play List Preferences tab is shown below:

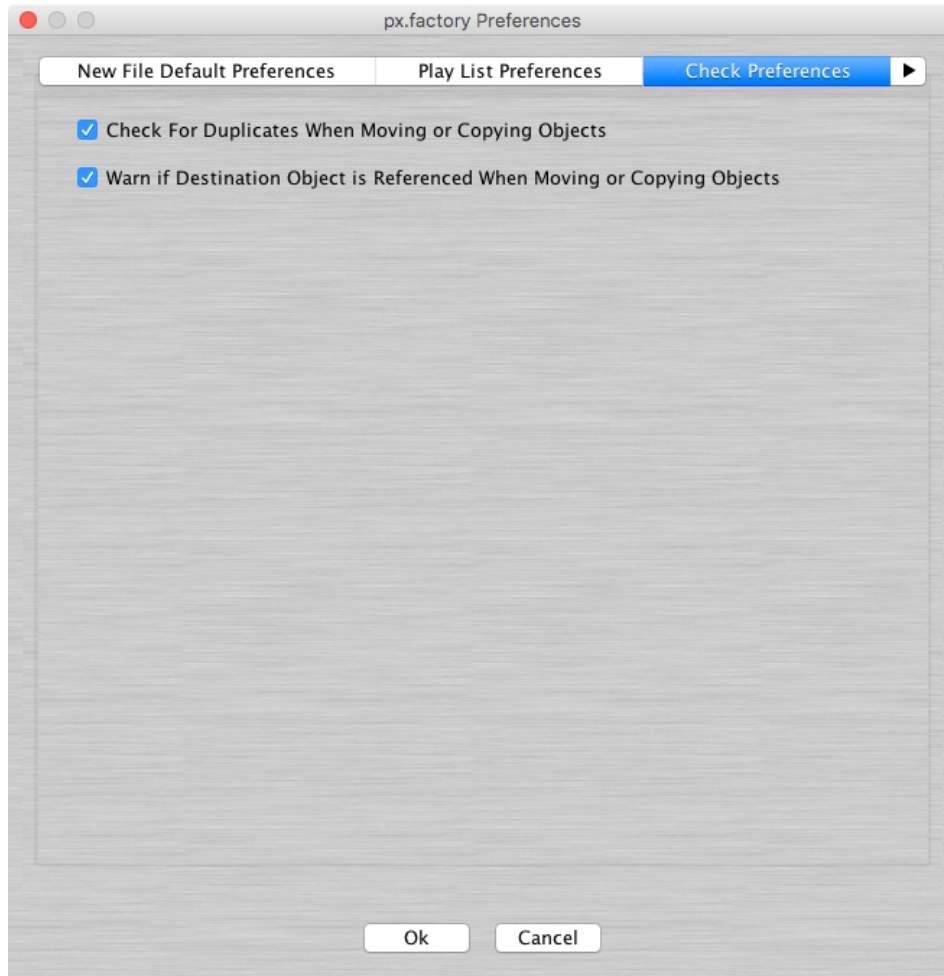


This tab has the following options:

Update Play Lists After Program Move	If you move Programs within a file then the Play Lists that reference the updated Programs are updated to ensure that the correct references to the Wave Sequences are maintained.
Copy User Programs With Play List	If you move or copy a Play List from one file to another file, then any Programs referenced by the Play List are also copied to the destination file if the Programs are not already present in the destination file.

Check Preferences

The Check Preferences tab is shown below:

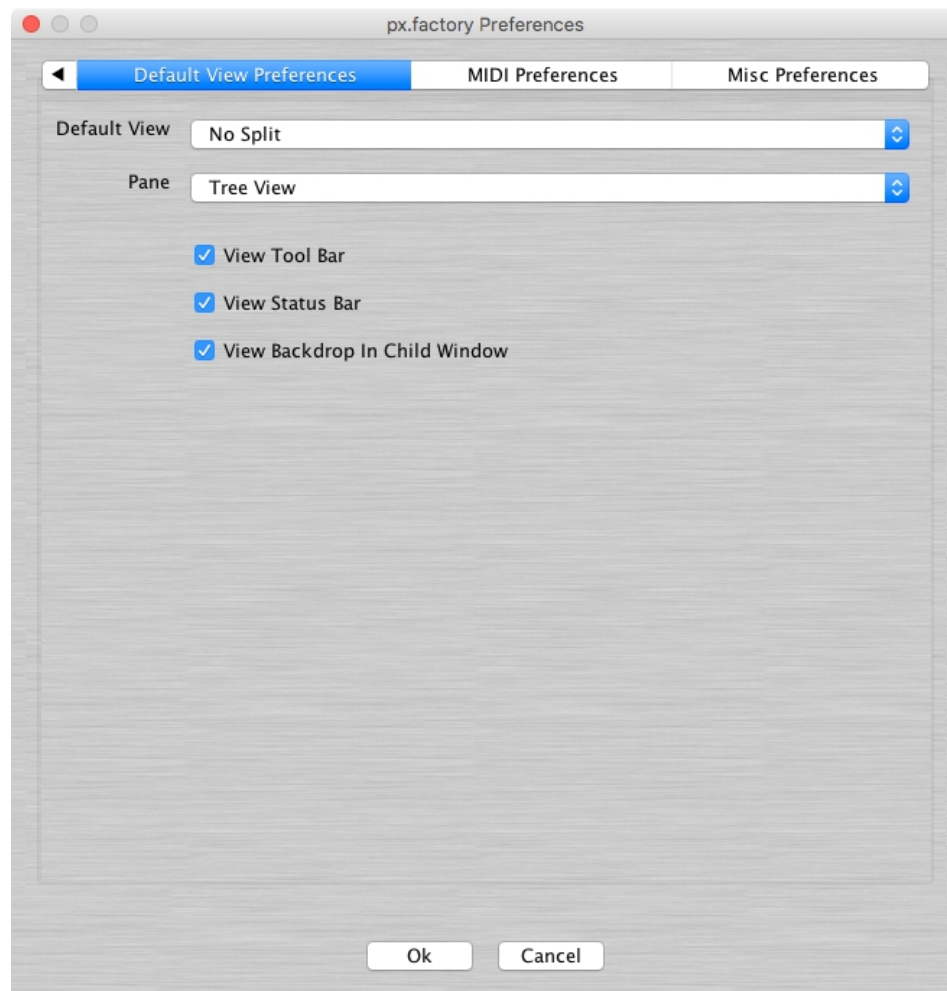


This tab has the following options:

Check For Duplicates When Moving or Copying Objects	If you move or copy an Object to a destination file and the Object already exists then a warning is presented giving you the option to continue or cancel the operation.
Warn if Destination Object is Referenced When Moving or Copying Objects	If you move or copy an Object and it is overwriting an Object that is being referenced, or you are deleting a referenced Object, then a warning is presented giving you the option to continue or cancel the operation.

Default View Preferences

The Default View Preferences tab is shown below:

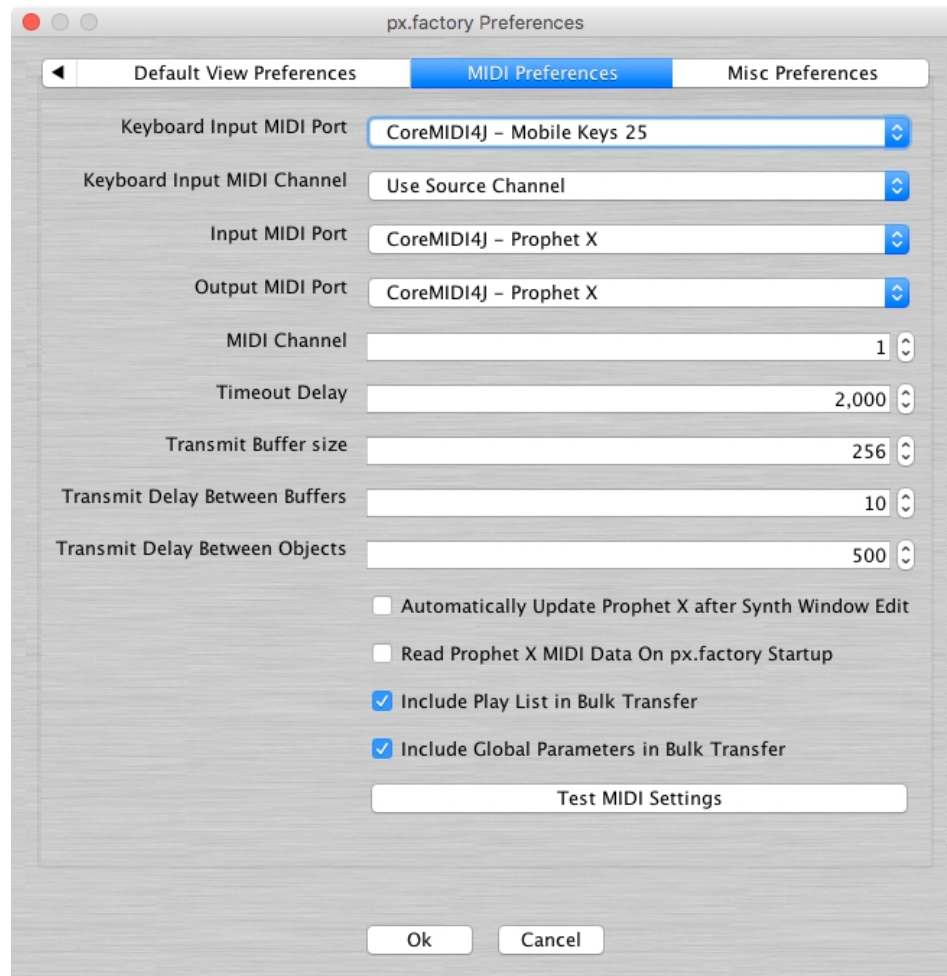


Default View	<p>Defines the type of view that is applied to a Child Window when it is first opened.</p> <p>The following options are available:</p> <ul style="list-style-type: none"> • No Split • Horizontal Split • Vertical Split
Pane/Left Pane/Top Pane	<p>Defines the type of view shown in the single pane of an unsplit view, the left pane of a horizontally split view or the top pane of a vertically split view.</p> <p>The following options are available:</p> <ul style="list-style-type: none"> • Tree View • Table View
Right Pane/Bottom Pane	<p>This control is only visible if the Default View option is set to either Horizontal Split or Vertical Split.</p> <p>Defines the type of view shown in the right pane of a horizontally split view or the bottom pane of a vertically split view.</p> <p>The following options are available:</p> <ul style="list-style-type: none"> • Tree View • Table View
View Tool Bar	Controls if the toolbar is visible at startup.
View Status Bar	Controls if the status bar is visible at startup.
View Backdrop In Child Window	This option controls if the image backdrop is displayed in Child Windows.

You may wish to turn off this option if you find the scrolling of the Child Window is slow on your machine.

MIDI Preferences

The MIDI Preferences tab is shown below:



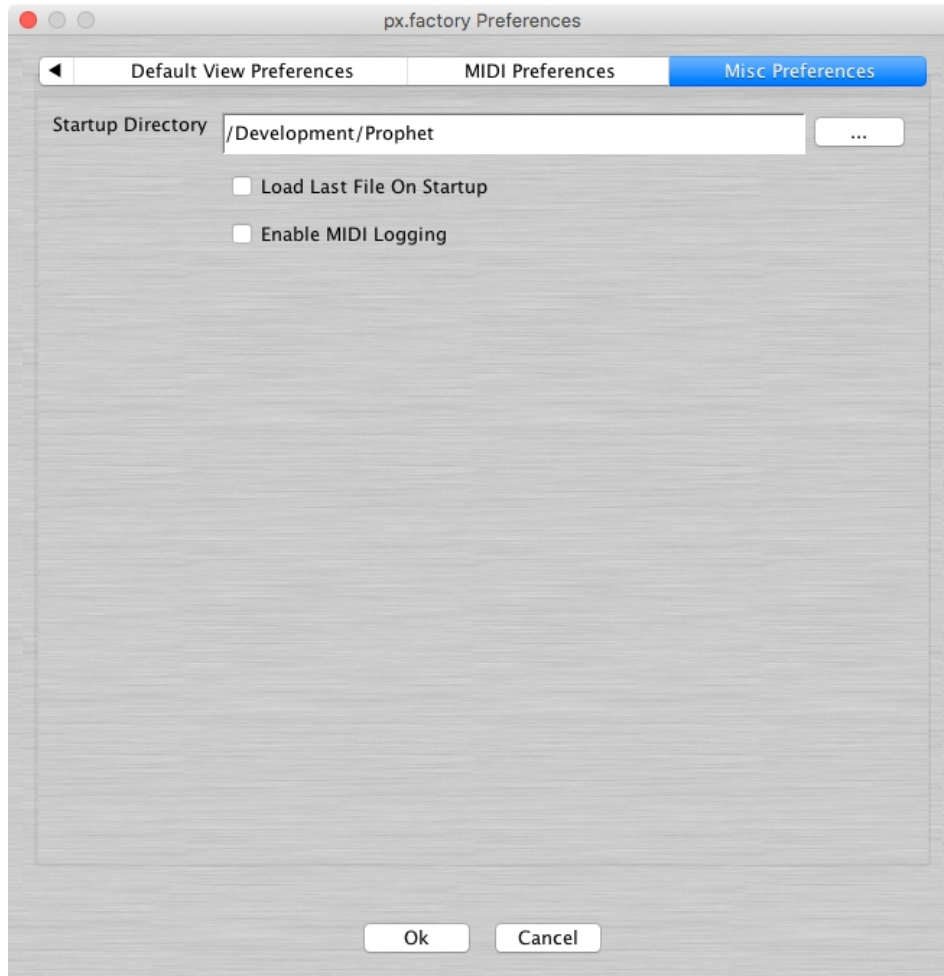
This tab has the following options:

Keyboard Input MIDI Port	Sets the port used for accepting MIDI data from a keyboard and routing it to the output MIDI port using the MIDI Router Dialog .
Keyboard Input MIDI Channel	Sets the channel to be used for transmitting MIDI data using the MIDI Router Dialog . The first option Use Source Channel in the drop down list, means that the MIDI channel from the source device is not altered.
Input MIDI Port	Sets the port used for MIDI input from the available ports.
Output MIDI Port	Sets the port used for MIDI output from the available ports.
MIDI Channel	Sets the channel number used for SYSEX transfers. This must match the Device No value set in the Prophet X's MIDI pages. If the Prophet X Channel No is set to all then it will receive all Prophet X SYSEX messages irrespective of the device number in the message.
Timeout Delay	Sets the time that px.factory waits for a response from the Prophet X before timing out and reporting an error. The value is specified in milliseconds.
Transmit Buffer Size	Sets the size of the transmit buffer used for sending data to the Prophet X. Some MIDI interfaces don't like data coming in big chunks and you may need to set this to get a reliable transfer of data. The value is specified in milliseconds.
Transmit Delay Between Buffers	This sets the delay that px.factory waits for between sending buffer sized data packets. The value is specified in milliseconds.
Transmit Delay Between Objects	This sets the delay that px.factory waits for after sending a complete object. The value is specified in milliseconds.
Automatically Update Prophet X after Synth Window Edit	When this option is checked, if the Prophet X Synthesizer Window is edited then the edited data will be automatically sent to the Prophet X.
Read Prophet X MIDI Data On px.factory Startup	When this option is checked the Programs and Play Lists (if enabled) are read from the Prophet X during the px.factory startup phase.
Include Play Lists In Bulk Transfer	When this option is checked, the Play Lists are included in bulk transfers. You may wish to uncheck this option if you don't use Play Lists.

Include Global Parameters In Bulk Transfer	When this option is checked, the Global Parameters are included in bulk transfers. You may wish to uncheck this option if you don't wish to store Global Parameters in your files.
Test Prophet X MIDI Settings	<p>Tests to see if MIDI communications is working using the specified values. The test:</p> <ol style="list-style-type: none"> 1. Sends Universal SYSEX Device Inquiry message on the specified MIDI output port 2. Waits for a response on the specified MIDI input port. If the response times out an error message is displayed 3. Checks to see if the expected response is received. If not then an error message is displayed

Miscellaneous Preferences

The Miscellaneous Preferences tab is shown below:




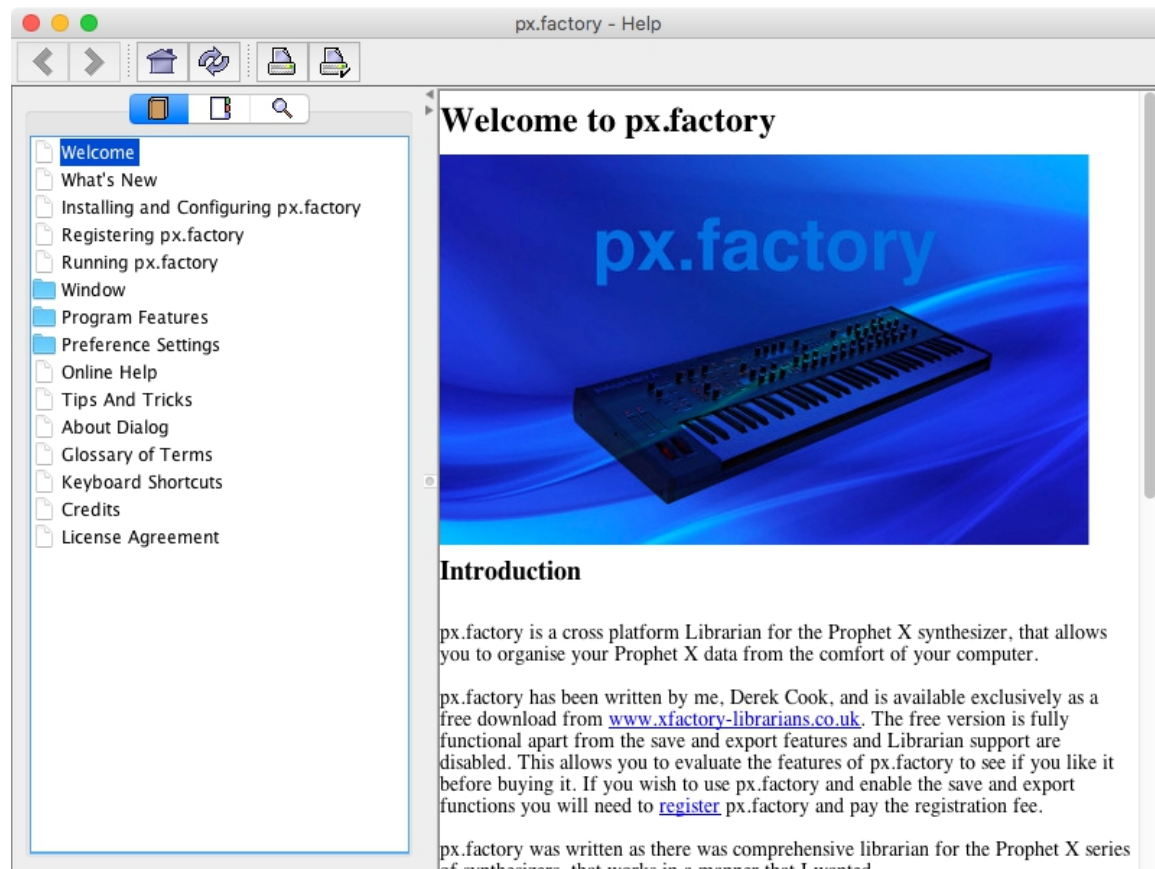
This tab has the following options:

Startup Directory	Allows the startup directory to be specified. Note: no validation is performed on this pathname, so please ensure it is valid. I may add validation later. The [...] button to the right of the field brings up a dialog that allows you to select the directory instead of typing it in.
Load Last File On Startup	Controls if the last file opened in the previous session is automatically loaded when kronos.factory is next invoked.
Enable MIDI Logging	When checked, MIDI communications is logged to the px.factory log file. Only turn this on when you need to, as the amount of data being written to the logs will slow down MIDI operations. This option is reset everytime px.factory starts, so you need to enable it in each session if you wish to use it.

Online Help

px.factory provides a full online version of the px.factory manual.




If you select **Help ...** from the [Help Menu](#) or the  button on the [Tool Bar](#), the Help Window is opened.









The Help Window works just like a web browser window, so its operation should be quite familiar to you.

The left pane of the Help Window shows a tabbed navigation pane to allow you to quickly navigate to a topic, whilst the right hand pane shows the help topic.

The navigation pane can show three different views, depending on which tab you select:


-  Displays the Table of Contents Navigation Pane, providing a structured list of px.factory topics
-  Displays the Index Navigation Pane, providing an indexed list of px.factory topics
-  Displays the Search Navigation Pane, allowing you to search the help topics for keywords

The Help Window has the following Tool Bar buttons:

-  Navigates backwards to the last page in the access sequence. If you press and hold the mouse button over this button, you will get a list of accessed pages
-  Navigates forwards to the next page in the access sequence. If you press and hold the mouse button over this button, you will get a list of accessed pages
-  Selects the home page
-  Refreshes the current page
-  Prints the current page
-  Displays the print settings dialog

Context Sensitive Help

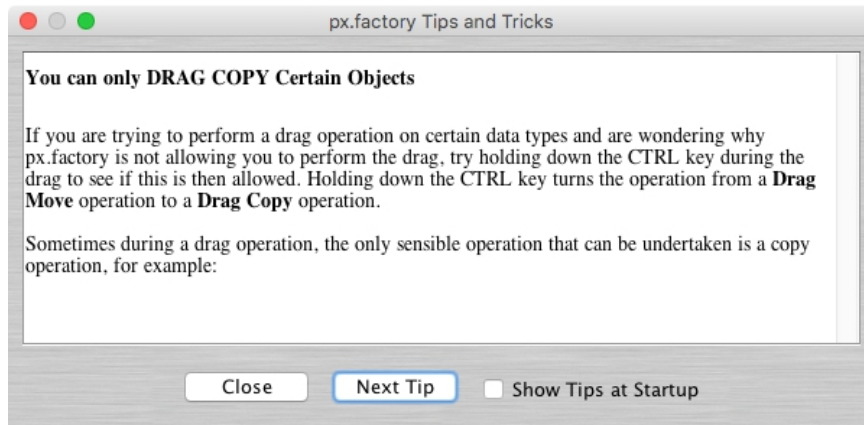
You can access context sensitive help by:

- pressing the your Computer Platform's **Help Key** (**F1** on the PC platform), which will load the help page associated with the Component within the px.factory [Window](#) that currently has **Focus** (is responding to the keyboard and mouse).
- pressing the  button on the [Tool Bar](#) and the clicking over the item of interest

You can press the **Help Key** in any px.factory Dialog to access the relevant help page for the Dialog.

Tips and Tricks

The px.factory tips and tricks dialog is shown below:



This dialog has the following controls:

Close	Closes the Tips and Tricks Dialog
Next Tip	Selects the next tip in sequence
Show Tips at Startup	If checked then when px.factory starts up, the Tips and Tricks Dialog is automatically opened

List of Tips and Tricks

Just in case you want the tips and tricks shown by the dialog as one single list, here they are.

Context Sensitive Help

If you press the **Help Key** (**F1** on a PC) then you can get context sensitive help, including within Dialogs.

Keyboard Shortcuts

px.factory supports numerous keyboard shortcuts as an alternative to using the mouse.

Take a look in the online help or the px.factory manual for details of the supported keyboard shortcuts

Object Colouring

If an Object such as a Wave or Program is coloured **blue**, then it is being referenced by another Object.

If you right click over a **blue** Object you'll find a "**List References**" context menu, which will show you the Objects that reference this one.

Find/Find Next Category Tool Bar Buttons

If you hold the **SHIFT** key down whilst pressing the **Find** or **Find Next** tool bar button the operations performed change from:

- **Find** to **Find Category**
- **Find Next** to **Find Next Category**

Table Sorting

You can sort Tables by clicking in the Table Column headers.

For example, if you click in the Name Column then the Table will be sorted by Name

You can only DRAG COPY Certain Objects

If you are trying to perform a drag operation on certain data types and are wondering why px.factory is not allowing you to perform the drag, try holding down the CTRL key during the drag to see if this is then allowed. Holding down the CTRL key turns the operation from a **Drag Move** operation to a **Drag Copy** operation.

Sometimes during a drag operation, the only sensible operation that can be undertaken is a copy operation, for example:

Copying/Moving Objects to a Different File

When you copy an Object to a different file then any Objects it references can also be copied at the same time.

For example if you move a Play List entry to a new File, then the referenced Program is also copied (assuming they are present within the file):

I think you'll agree this is a cool feature!

This feature can be selectively turned off for different Object types if required within the Preferences Dialog.

Workspaces

px.factory allows you to save a Window arrangement as a **Workspace** which you can then recall at a later date.

Moving Objects within a File

When you move an Object within a file then any Objects that reference the Object type being moved can be updated to ensure they all still reference the correct objects after the move.

For example if you move a Program within a file then Play List is updated

I think you'll agree this is a cool feature!

This feature can be selectively turned off for different Object types if required within the Preferences Dialog.

About px.factory

The about dialog is shown when the **About ...** option from the [Help Menu](#) or the ⓘ button on the [Tool Bar](#) is selected.



The dialog shows some basic information about the program including the version number and the [License](#) status of px.factory.

It also shows which Java Run Time Environment that px.factory is running in.

Glossary

Term	Definition
------	------------

Keyboard Short Cuts

px.factory provides many keyboard shortcuts for navigation and performing editing operations.

In general, navigating between components uses these keys:

- **Tab.** Moves keyboard focus to the next component or to the first member of the next group of components.
- **Ctrl-Tab.** Moves keyboard focus to the next component or to the first member of a group of components when the current component accepts a tab (as in text fields, tables, and tabbed panes).
- **Shift-Tab.** Moves keyboard focus to the previous component or to the first component in the previous group of components.
- **Arrow keys.** Move keyboard focus within the individual components of a group of components--for example, within menu items in a menu or within tabs in a tabbed pane.

The shortcuts are divided into the following categories:

- [Keyboard shortcuts for the Desktop and Child Windows](#)
- [Keyboard shortcuts for Tree Views](#)
- [Keyboard shortcuts for Table Views](#)
- [Keyboard shortcuts for Editing within Tree and Table Views](#)
- [Keyboard shortcuts for the Preferences Dialog](#)
- [Keyboard shortcuts for Text Editing Fields](#)

Keyboard Shortcuts for the Desktop and Child Windows

Action	Keystroke	Notes
New File	Ctrl-N	
Open File	Ctrl-O	
Save File	Ctrl-S	
Save File As	Ctrl-A	
Save All Files	Ctrl-L	
Open Workspace	Ctrl-ALT-O	
Save Workspace	Ctrl-ALT-S	
Tile Horizontally	Ctrl-H	
Tile Vertically	Ctrl-V	
Maximise Windows	Ctrl-M	
Restore Windows	Ctrl-R	
Minimise Windows	Ctrl-I	
Closes Child Window	Ctrl-F4	
Moves Child Window	Ctrl-F7	
Resizes Child Window	Ctrl-F8	
Minimizes Child Window	Ctrl-F9	
Switch to next Child Window on the Desktop	Ctrl-F6	

Keyboard Shortcuts for Tree Views

Action	Keystroke	Notes
Expands current Object	Right arrow	
Collapses current Object, or moves focus to Parent Object	Left arrow	
Moves focus up one Object	Up arrow	
Moves focus down one Object	Down arrow	
Moves focus to first Object in Tree View	Home	
Moves focus to last Object in Tree View	End	
Moves up one view	Page Up	
Moves down one view	Page Down	
Selects all Object in Tree View	Ctrl-A, Ctrl-/	
Deselects all Object in Tree View	Ctrl-\	
Extends selection down	Shift-down arrow	
Extends selection up	Shift-up arrow	
Extends selection to beginning of Tree View	Shift-Home	
Extends selection to end of Tree View	Shift-End	
Extends selection up one view	Shift-PgUp	
Extends selection down one view	Shift-PgDn	

Keyboard Shortcuts for Table Views

Action	Keystroke	Notes
Moves up one row	Up arrow Shift-Return	
Moves down one row	Down arrow Return	
Scrolls up one view	Page Up	
Scrolls down one view	Page Down	
Moves focus and view to first cell in the current row	Home	
Moves focus and view to last cell in the current row	End	
Moves focus and view to first cell in the current column	Ctrl-Home	
Moves focus and view to last cell in the current column	Ctrl-End	
Allows editing in a Category cell	F2, Down Arrow	
Allows editing in a Name cell.	F2	
Resets cell to the state it was in before it was edited	Escape	
Selects entire table	Ctrl-A	
Extends selection up one row	Shift-up arrow	
Extends selection down one row	Shift-down arrow	
Extends selection to beginning of table	Ctrl-Shift-Home	
Extends selection to end of table	Ctrl-Shift-End	
Extends selection up one view	Shift-PgUp	
Extends selection down one view	Shift-PgDn	

Keyboard Editing Short Cuts for both Tree and Table Views

Action	Keystroke	Notes
Undo	Ctrl-Z	
Redo	Ctrl-Y	
Cut	Ctrl-X	
Copy	Ctrl-C	
Paste	Ctrl-V	
Find	Ctrl-F	
Find Next	F3	
Find Category	Ctrl-Shift-F	
Find Next Category	Shift-F3	
Goto	Ctrl-G	
Insert Object	INS	Not allowed on Collections
Delete Object	DEL	Not allowed on Collections
Rename Object	F4	Single Named Object Only
Initialise Object	F5	
List References	F6	Single Named Object Only
Map To Flash	F7	
Map To Ram	Shift-F7	
Export to MIDI Edit Buffer	F8	Single Voice or Performance Only
Export to MIDI Memory	Shift-F8	Voices or Performances Only
Import from MIDI Edit Buffer	F9	Single Voice or Performance Only
Import from MIDI Memory	Shift-F9	Voices or Performances Only

Keyboard Shortcuts for the Preferences Dialog

Action	Keystroke	Notes
Navigates through tabs	Arrow keys	
Moves from tab to its associated content pane	Ctrl-down arrow	
Moves from content pane to its associated tab	Ctrl-up arrow	

Keyboard Shortcuts for Text Editing Fields

Action	Keystroke	Notes
Moves insertion point one character to the right	Right arrow	
Moves insertion point one character to the left	Left arrow	
Moves insertion point to beginning of next word	Ctrl-right arrow	
Moves insertion point to beginning of previous word	Ctrl-left arrow	
Moves insertion point to beginning of field	Home	
Moves insertion point to end of field	End	
Submits text entry	Enter, Return	
Extends selection to beginning of line	Shift-Home	
Extends selection to end of line	Shift-End	
Extends selection one character to the left	Shift-left arrow	
Extends selection one character to the right	Shift-right arrow	
Extends selection to next word	Shift-Ctrl-right arrow	
Extends selection to previous word	Shift-Ctrl-left arrow	

Credits

Thanks go, in no particular order, to:

- Sequential for making such a fantastic synthesizer in the Prophet X

3rd party software used in px.factory

The following 3rd part software is used to build px.factory and provide some program features.

CoreMIDI4J License

px.factory makes use of [CoreMIDI4J](#) to provide MIDI support on OS X, which is issued under the [EPL 1.0 License](#).

Table Filter License

px.factory makes use of [TableFilter](#) by [Luis M. Pena](#), which is issued under the [MIT License](#).

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache Derby License

px.factory makes use of [Apache Derby](#) to provide the library function, which is issued under the [Apache 2.0 License](#).

Apache Log4J License

px.factory makes use of [Apache Log4J](#) to provide logging services, which is issued under the [Apache 2.0 License](#).

Dom4J License

px.factory makes use of [Dom4J](#) to provide XML services, which is issued under the [Maplesoft Dom4J License](#).

THIS SOFTWARE IS PROVIDED BY METASTUFF, LTD. AND CONTRIBUTORS "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL METASTUFF, LTD. OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

px.factory makes use of [IzPack](#) to provide the px.factory installer, which is issued under the [Creative Commons 4.0 License](#).

Java Wizard License

The px.factory Library Batch Importer makes use of the [Java Wizard Project](#) which is distributed under the [Open Source Initiative Common Development and Distribution License](#).

I have adapted the Wizard software to allow me to display two progress bars in the Wizard. If anybody wishes to have the source code for how I did this, please email me at supportx@xfactory-librarians.co.uk and I will happily provide the adapted source code.

px.factory License

The unregistered version of px.factory allows you to evaluate the features of the program, which are fully functional except the File Save, MIDI Export and Librarian features.

However, px.factory cannot be included as part of a commercial distribution or on other web sites without the express permission of the Author. I am happy for you to provide links to www.xfactory-librarians.co.uk, to maximise public access to px.factory.

px.factory license files are issued to users who have registered px.factory, to gain access to the File Save, MIDI Export and Librarian features. If you have been issued with a license key you are not allowed to distribute a license key to other users. Please respect this, and the amount effort I have put into making this application.

You are permitted to install and activate px.factory on two computers, for which the license is valid.

If you wish to install and activate px.factory on more than two computers, then an additional license can be purchased at 50% of the cost of the main license.

The Author retains the copyright on the program and all of its originally developed features.

Disclaimer

Like all software products, this software is provided "AS IS", and without any warranties expressed or implied. But hey, remember, you're getting it for a very low cost!

I have worked hard to ensure this software works correctly, as I certainly don't want it trashing my data. However, I cannot be held liable for any problems arising as a consequence of you using this software on your computer and your data. You use this product at your risk.

Like all new products, please proceed and evaluate with care, and ensure that all your important data files are backed up before installation and evaluation.

Having gotten that out of the way, if you do encounter any problems, please submit a bug report to support@xfactory-librarians.co.uk, and I will endeavour to provide a fix for any reported problems.

Licensing Conditions for 3rd party software

Apache Derby License

px.factory makes use of [Apache Derby](#) to provide the library function, which is issued under the [Apache 2.0 License](#).

Apache Log4J License

px.factory makes use of [Apache Log4J](#) to provide logging services, which is issued under the [Apache 2.0 License](#).

CoreMIDI4J License

px.factory makes use of [CoreMIDI4J](#) to provide MIDI support on OS X, which is issued under the [EPL 1.0 License](#).

Dom4J License

px.factory makes use of [Dom4J](#) to provide XML services, which is issued under the [Maplesoft Dom4J License](#).

THIS SOFTWARE IS PROVIDED BY METASTUFF, LTD. AND CONTRIBUTORS ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL METASTUFF, LTD. OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Java Wizard License

The px.factory Library Batch Importer makes use of the [Java Wizard Project](#) which is distributed under the [Open Source Initiative Common Development and Distribution License](#).

Table Filter License

px.factory makes use of [TableFilter](#) by [Luis M. Pena](#), which is issued under the [MIT License](#).

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Tritonus License

px.factory makes use of [Tritonus Java Sound Extensions](#), which is issued under the [GNU Lesser General Public License](#).