MT LAUNCHER 1.1

INSTALLATION MANUAL

Revision 2
February 2018

Copyright © 2018 MultiTaction. All rights reserved.

This manual is intended for the owners and operators of MT Showcase. It contains guidelines for the proper usage of the product. Information in this manual is subject to change without prior notice to product owners. For the latest product details and guidelines please visit the product website.

Trademarks

Contents

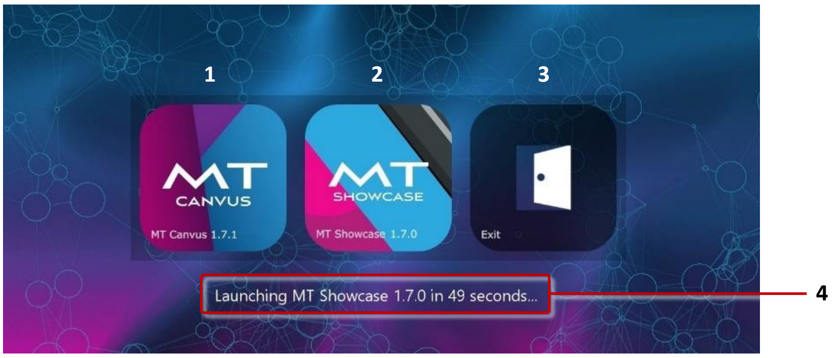
1 Introduction

This manual describes how to install MT Launcher 1.1.

MT Launcher is a tool for launching products such as MT Canvus and MT Showcase. It displays as a menu on your video wall and comprises a series of tiles that users can tap to launch the required product.

When used in conjunction with features such as the Exit Showcase widget in MT Showcase, MT Launcher provides a quick and easy method for switching between MultiTaction products. It also provides logging functionality and monitors product processes, restarting them if they crash.

MT Launcher is included in the MultiTaction base image. Other customers must install MT Launcher manually. You must then configure MT Launcher plus any *pipelines* that you want to include.



MT Launcher menu. This example shows tiles for (**1**) *MT Canvus,* (**2**) *MT Showcase and* (**3**) *Exit tile. If a product is configured to launch automatically when MT Launcher starts, a countdown timer* (**4**) *shows how long before the product launches.*

Note: *The Exit tile is displayed by default. However, you can hide this tile, if required. (MT Launcher is designed to provide end-users with a simple method for launching products. You may not want to provide end-users with a simple button to exit to the desktop from MT Launcher.)*

1.1 What is a pipeline?

A *pipeline* refers to the set of processes that are collectively required to launch a MultiTaction product such as MT Showcase or MT Canvus. These MultiTaction products now have built-in compatibility with MT Launcher and ship with the configuration files required by MT Launcher.

1.2 Deployment summary

Deploying MT Launcher involves the following steps:

1. **Install MT Launcher**

   On Ubuntu application computers, you can install MT Launcher from an installation package or by using the `apt` utility. On Windows application computers, you must download and run the MT Launcher installer. See section 2.

2. **Configure MT Launcher**

   MT Launcher is designed to operate out-of-the-box, with minimal configuration. But if you want to customize features or behavior, you can configure MT Launcher by editing custom versions of the main configuration file, launcher.ini, and the pipeline configuration files for individual products. See section 3.

3. **Restart MT Launcher**

   You must restart MT Launcher for the configuration changes to take effect.
   See section 4.

1.3 MultiTaction support

If you need technical assistance, please contact MultiTaction Support:
https://www.multitaction.com/support-services

2 Install MT Launcher

2.1 Install on Ubuntu application computers

If you installed the MultiTaction base image, MT Launcher is installed automatically on the application computer. You now need to configure MT Launcher; skip this chapter and continue directly to section 3.

If you did *not* install the base image, you can manually install MT Launcher from an installation package or by using the `apt` utility. These tasks are described below.

2.1.1 Install from an installation package

Skip this task if you installed the MultiTaction base image.

To install MT Launcher from an installation package, follow these steps:

1. Browse to the MultiTaction Downloads page:
   a. Register on the MultiTaction Cornerstone web site:
      https://cornerstone.multitouch.fi/
   b. Contact MultiTaction Sales and request access to the Downloads page:
      https://www.multitaction.com/support-services
   c. Launch a browser on the application computer and log on to the MultiTaction Cornerstone web site (see step 1.a).
   d. Browse to the Downloads page:
      https://cornerstone.multitouch.fi/mt-launcher-downloads

2. Download the MT Launcher installer onto the application computer. The installer filename is similar to this example:
   mt-launcher-1.0.0-build6800-Ubuntu-14.04-amd64.sh

3. Run this command to execute the MT Launcher installation script:
   ```
   $ sudo sh <installer>
   ```
   Where `<installer>` is the file you downloaded in the previous step. For example:
   ```
   $ sudo sh mt-launcher-1.0.0-build6800-Ubuntu-14.04-amd64.sh
   ```

2.1.2 Install using the apt command

Skip this task if you installed the MultiTaction base image.

To install MT Launcher using Ubuntu's `apt` command line utility, follow these steps:

1. Right-click the desktop on the application computer and launch a terminal emulator.

2. Run this command to enable the mt-software-stable repository list definition file:
   ```
   $ echo "deb [arch=amd64]
     http://update.multitouch.fi/mt-stable stable main" |
     sudo tee /etc/apt/sources.list.d/mt-software-stable.list
   ```

3. Run the following command to set appropriate read and write permissions for the mt software-stable repository list definition file:
   ```
   sudo chmod 644 /etc/apt/sources.list.d/mt-software-stable.list
   ```

4. Run this command to update the list of available installation packages:
   ```
   $ sudo apt-get update
   ```

5. Run this command to download and install MT Launcher:
   ```
   $ sudo apt-get install mt-launcher
   ```

2.2 Install on Windows application computers

To install MT Launcher, follow these steps:

1. Browse to the MultiTaction Downloads page:
   a. Register on the MultiTaction Cornerstone web site:
   https://cornerstone.multitouch.fi/
   b. Contact MultiTaction Sales and request access to the Downloads page:
   https://www.multitaction.com/support-services
   c. Launch a browser on the application computer and log on to the MultiTaction Cornerstone web site (see step 1.a).
   d. Browse to the Downloads page:
   https://cornerstone.multitouch.fi/mt-launcher-downloads

2. Download the MT Launcher installer onto the application computer. The installer filename is similar to this example:
   mt-launcher-1.0.0-build6800.exe

3. Run the MT Launcher installer.

4. When the MT Launcher Setup Wizard launches:
   a. Choose the installation folder.
   b. Step through the wizard screens and click Install.

5. An MT Launcher shortcut is added to the Windows desktop:

3 Configure MT Launcher

MT Launcher is designed to operate out-of-the-box, with minimal configuration. But if you want to customize features or behavior, you can configure MT Launcher by editing custom versions of the main configuration file, launcher.ini, and the configuration files for individual product pipelines (see section 1.1).

3.1 Default and custom versions of the configuration files

MT Launcher supports default and custom versions of launcher.ini and pipeline configuration files. If you want to configure MT Launcher, you must create and edit custom versions of these files. (For the location of these files, see section 3.1.2 and section 3.1.3.)

- **Default versions** of these configuration files are installed with MT Launcher into a system folder. If you upgrade or reinstall MT Launcher, the default versions are overwritten.

  If MT Launcher cannot detect any custom versions (see below), it uses settings in the default versions. If it does detect a custom version, it only uses the custom version and disregards the default version.

  Do not edit these default versions!

- **Custom versions** of these configuration files override the default versions. To create a custom version, copy the default version into your user folder. Then edit the new custom version. Settings in custom versions are retained, even if you upgrade or reinstall MT Launcher. Custom versions are also user-specific.

  When you configure MT Launcher, you must always edit the custom versions!

Note: *Custom versions of launcher.ini and pipeline configuration files always take precedence. If MT Launcher cannot detect a custom version in your user folder, it will instead use the default version in the system folder.*

3.1.1 Must a custom configuration file contain all the settings?

Yes. If you create a custom version of launcher.ini, MT Launcher ignores the default version completely and only refers to settings in the custom version. For this reason, the custom version of launcher.ini must contain all the settings that MT Launcher needs to operate. The same requirement also applies to pipeline configuration files.

For example, if you only want to customize the `timeout` setting, you must still include the full range of settings in your custom launcher.ini. If you omit the other settings, MT Launcher will revert to using hard-coded default values for any missing settings.

3.1.2 Where is launcher.ini?

The main configuration file for MT Launcher is launcher.ini.

Ubuntu application computers

- **Default version:** Find the default launcher.ini in this system folder:
  /etc/MultiTaction/launcher
- **Custom version:** To create a custom launcher.ini, copy the default version to:
  ~/MultiTaction/launcher

  Where ~/ refers to the home folder of the *runtime user*. If the user logged on while MT Launcher runs is 'multi', the expanded path is:
  /home/multi/MultiTaction/launcher

  You do not need to modify the file permissions when you copy the default launcher.ini. The default launcher.ini has a 644 permission.

Windows application computers

- **Default version:** Find the default launcher.ini in this system folder:
  %PROGRAMDATA%\MultiTaction\launcher

  The expanded path is C:\ProgramData\MultiTaction\launcher.
- **Custom version:** To create a custom launcher.ini, copy the default version to:
  %APPDATA%\MultiTaction\launcher

  Where %APPDATA% refers to the *runtime user's* profile. If the user logged on while MT Launcher runs is multi, the expanded path is:
  C:\Users\multi\AppData\Roaming\MultiTaction\launcher

  You do not need to modify the file permissions when you copy the default version of launcher.ini.

3.1.3 Where are the pipeline configuration files?

MT Launcher uses pipeline configuration files to configure the launch of individual programs.

Ubuntu application computers

- **Default version:** Find the default pipeline configuration file in this system folder: /etc/MultiTaction/launcher/apps
- **Custom version:** To create custom versions of pipeline configuration files, copy the default versions into:
  ~/MultiTaction/launcher/apps

  Where ~/ refers to the home folder of the *runtime user*. If the user logged on while MT Launcher runs is 'multi', the expanded path is:
  /home/multi/MultiTaction/launcher/apps

  You do not need to modify the file permissions when you copy the default pipeline configuration files. The default pipeline configuration files have a 644 permission.

Windows computers

- **Default version:** Find the default pipeline configuration file in this system folder: %PROGRAMDATA%\MultiTaction\launcher\apps

  The expanded path is C:\ProgramData\MultiTaction\launcher\apps.
- **Custom version:** To create custom versions of pipeline configuration files, copy the default versions into your user folder:
  %APPDATA%\MultiTaction\launcher\apps

  Where %APPDATA% refers to the *runtime user's* profile. If the user logged on while MT Launcher runs is multi, the expanded path is:
  C:\Users\multi\AppData\Roaming\MultiTaction\launcher

  You do not need to modify the file permissions when you copy the default pipeline configuration files.

3.2 Edit the custom version of launcher.ini

MT Launcher is designed to operate out-of-the-box, with minimal configuration.
But if you want to configure MT Launcher operations, you must edit a *custom version* of launcher.ini.

3.2.1 Edit MT Launcher's general settings

Follow these steps:

1. Edit your custom version of launcher.ini; see section 3.1.2.
2. Edit the following settings in the `[ui]` section:

```
[ui]
show-version=<true or false>
allow-exit=<true or false>
show-invalid-tiles=<true or false>
location=<x y>
width=<n>
expected-launch-duration=<n>
```

   See below for settings descriptions.

3. After editing launcher.ini, restart MT Launcher for the changes to take effect; see section 4.

Where:

- `show-version` specifies that MT Launcher displays its version number. By default, this setting is `true`.
- `allow-exit` determines whether MT Launcher includes an Exit tile (see the screenshot on on page 4). This tile is for administrators and allows them to exit MT Launcher and access the OS desktop.

  By default, this setting is `true`. If you want to disable the Exit tile to prevent end-users exiting to the desktop from MT Launcher, set the setting to `false`.

  Notes

  - *If you installed MT Launcher as part of the MultiTaction base image, the Exit tile is disabled by default.*
  - *Whether the Exit tile is enabled or disabled, a keyboard shortcut is always available for administrators to exit MT Launcher; see section 4.*

- `show-invalid-tiles` specifies whether to show product tiles for *invalid pipelines*. By default, this setting is `false`.

  In practice, you rarely need to edit this setting. For example, you may temporarily set the setting to `true` if instructed to do so by MultiTaction support staff or if you are developing and testing a new MT Launcher pipeline.

  Note: *An 'invalid pipeline' refers to a product that still has a pipeline configuration file but which failed one or more validation tests. For example, a pipeline can be invalid if MT Launcher cannot find a valid executable file or if the specified working directory does not exist. Such conditions can apply to older, superseded versions of a product following an upgrade.*

- `location` specifies the location of the center point of the MT Launcher menu on the screen. Use this setting to relocate menu. For example, you may want to avoid a Cell bezel or, on a larger video wall, make the tiles easier to reach by moving MT Launcher lower down the screen.

  `<x y>` are the horizontal and vertical coordinates of the menu's center point. You can specify coordinates as percentages of the screen width and height, or in pixels, centimeters or inches. If you omit the units, the values default to pixels. By default, the location is 50% 50%, locating MT Launcher in the center of the screen.

  For example, to display MT Launcher in the lower left corner of the MultiTaction Meeting Room screen (comprising three Cells in portrait mode), you can specify any of the following:

```
location=16% 84%
```

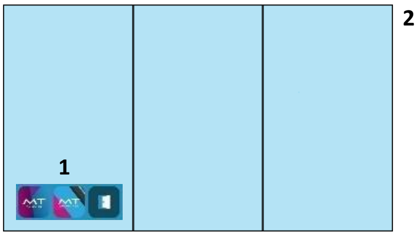
```
location=540 1600
```

```
location=540px 1600px
```

```
location=34cm 100cm
```

```
location=13in 39in
```

  Note: *All measurements are from the top-left corner of the screen.*



*Example MT Launcher menu (**1**) in lower left corner of MultiTaction Meeting Room screen (**2**)*

- `width` specifies the width of the MT Launcher menu on the desktop, where `<n>` specifies a value in pixels, centimeters or inches. To specify the unit, add `px`, `cm` or `in` respectively. If you omit the units, the value defaults to pixels. By default, the width is 1000px, wide enough for 3 columns of product tiles.

  For example, to restrict the MT Launcher width to 2 columns of tiles, you can specify any of the following:

```
width=600
```

```
width=600px
```

```
width=40cm
```

```
width=16in
```

  Note: *All measurements are from the top-left corner of the screen.*

- `expected-launch-duration` defines the *launch period*. This is the amount of time, in seconds, that your products need to successfully start up.

  During a product's launch period, all further user interaction is suspended, a "Running..." advisory displays above the tile, and the MT Launcher background animation continues to run. When the launch period ends, background animation freezes to minimize CPU usage.

  The launch period applies to all product tiles and defaults to 3 seconds. You will never need to set a shorter launch period, but you may need to set a longer launch period if some products take significantly longer than 3 seconds to launch:

  - If you were to set a very short launch period (for example, zero seconds), the background animation would not cause high CPU usage but MT Launcher would appear to freeze before the product starts up. Users might find this distracting.
  - If you set a launch period that is longer than your products need (for example, 30 seconds), the background animation will continue to run after products start up, causing high CPU usage even though MT Launcher is not visible on-screen. This may affect product performance in the first few seconds. (The background animation will stop and CPU usage will return to normal when the launch period ends.)

  Note: *In previous versions of MT Launcher before this setting was available, when a user tapped a tile, there was no "Running..." advisory and the background animation froze immediately. If the product was slow to launch, users sometimes believed (incorrectly) that MT Launcher had crashed.*

3.2.2 Specify how often log files are compressed

You can specify a maximum age for uncompressed log files. When MT Launcher starts, any log files older than this maximum age are compressed to .gz files.

Follow these steps:

1. Edit your custom version of launcher.ini; see section 3.1.2.
2. Edit the log file compression setting in the `[log]` section:

   ```
   [logs]
   rotate-period=86400
   ```

   Where `rotate-period` specifies the maximum age of uncompressed log files, in seconds. The default is 86,400 (equivalent to 24 hours).

3. After editing launcher.ini, restart MT Launcher for the changes to take effect; see section 4.

For details about where MT Launcher log files are saved, see section 3.4.1.

3.2.3 Specify which product starts automatically

You can choose *one* of the available product tiles to launch automatically when MT Launcher starts. You can also set a delay before the tile starts.

Follow these steps:

1. Edit your custom version of launcher.ini; see section 3.1.2.

2. Edit the following settings in the `[autostart]` section:

   ```
   [autostart]
   tile=<pipeline configuration>
   timeout=<n>
   ```

   Where

   - `tile` identifies the product tile that starts automatically. Set `tile` to the base name of the product's pipeline configuration file (that is, the file name minus the .ini extension). For example, if the product's pipeline configuration file is mt-canvus-2.0.0.ini, set `tile` to:

     ```
     tile=mt-canvus-2.0.0
     ```

     If you do not want any product tile to start automatically, leave this setting undefined (null):

     ```
     tile=
     ```

     Note: *Only one product can start automatically! The* `tile` *setting must only specify a single product. It cannot specify a comma-separated list of products.*

   - `timeout` specifies a delay, in seconds, before the specified product starts automatically. A countdown timer in MT Launcher shows how long before the product starts (see **item 4** in the screenshot in section 1).

     The default delay is 60 seconds, but you can set a shorter delay. For example, if you want the product to start 15 seconds after MT Launcher starts, enter:

     ```
     timeout=15
     ```

     If you want the product to start immediately when MT Launcher starts, either leave this setting undefined (null) or set it to zero:

     ```
     timeout=
     timeout=0
     ```

3. After editing launcher.ini, restart MT Launcher for the changes to take effect; see section 4.

Now edit the custom pipeline configuration files; continue to section 3.3.

3.3 Edit the custom pipeline configuration files

MT Launcher is designed to operate out-of-the-box, with minimal configuration. But if you want to configure the launch of individual programs, you must create and edit a *custom version* of the relevant pipeline configuration files (section 3.3.1).

You can then:

- Configure the product tile in MT Launcher. You can customize the icon (image file) and the display name in a product tile. See section 3.3.2.
- (*Optional*) Configure a 'follow-on' product to launch when the current product exits; see section 3.3.3.
- Configure the *pipeline log file*. You can specify where pipeline logs are saved and you can set custom log file names. See section 3.3.4.
- Edit the process blocks. A process block specifies: which executable to run, the working directory for the process; arguments passed to the executable; and so on. See section 3.3.5.

3.3.1 Create a custom version of the pipeline configuration file

To create a *custom version* of a pipeline configuration file, you must copy the *default version* into the home folder of the runtime user (Ubuntu) or the runtime user's profile (Windows). For details about these file locations, see section 3.1.3.

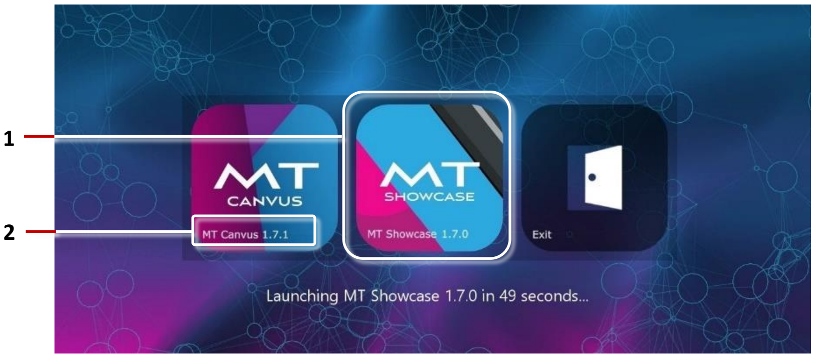
Example: The user logged on while MT Launcher runs is 'multi'. To create a custom pipeline configuration file for MT Canvus 1.7.0, you therefore need to copy the default version to here:

Ubuntu: /home/multi/MultiTaction/launcher/apps/mt-canvus-1.7.0.ini
Windows: C:\Users\multi\AppData\Roaming\MultiTaction\launcher\mt-canvus-1.7.0.ini

Note: *Pipeline configuration files are not the same as the product configuration files! For example, the MT Canvus configuration file mt-canvus.ini has a completely different purpose to mt-canvus-1.7.0.ini, the pipeline configuration file used by MT Launcher.*

3.3.2 Configure the product tile

You can customize the icon (image file) and the display name in a product tile.



MT Launcher. Product tile icon (**1**) and display name (**2**).

Follow these steps:

1. Edit the custom pipeline configuration file; see section 3.3.1.
2. Edit the following settings in the `[ui]` section:

```
[ui]
name=<name>
icon=<image file>
```

Where:

- `name` is the display name of the product as it appears on the product tile in MT Launcher. For example:

```
name=MT Canvus 1.7.0
```

If you want a tile to have no name (for example, because the tile is identified by a distinctive icon), you can leave this setting undefined (null).

- `icon` specifies the path and image file for the product tile in MT Launcher. *We strongly recommend that you specify an absolute path.*

Ubuntu example

```
icon=/opt/mt-canvus-1.7.0/CanvusAssets/Icons/canvus-icon.png
```

Windows examples

```
icon=C:/Program Files/MT Canvus/CanvusAssets/Icons/canvus-icon.png
icon=C:\\Program Files\\MT Canvus\\CanvusAssets\\Icons\\canvus-icon.png
```

Note: *Windows paths must use / forward slashes or \\ double backslashes. They cannot use \ single backslashes.*

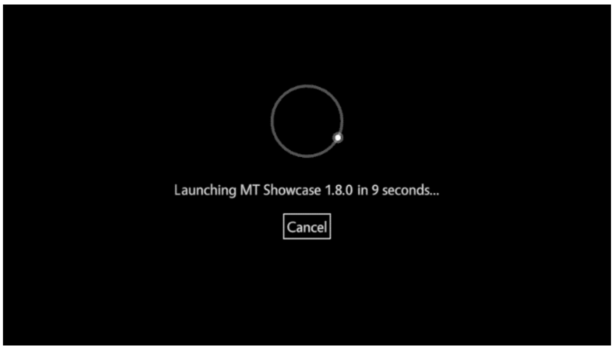
If you do specify a relative path, the path is relative to the MT Launcher working directory.

3. Restart MT Launcher for the pipeline changes to take effect; see section 4.

3.3.3 Launch a 'follow-on' product when the current product exits

If required, you can specify a 'follow-on' product to launch automatically when a user exits the current product. For example, you can specify that a MT Showcase app always launches when MT Canvus exits.

By default, the transition to the next product is immediate, but you can specify a pause (for example, 10 seconds). During this pause, MT Launcher displays a countdown timer and Cancel button, allowing users to cancel the next product and return to MT Launcher.



Launching MT Showcase 1.8.0 in 9 seconds...

Optional countdown timer and Cancel button for follow-on product

Follow these steps:

1. Edit the custom pipeline configuration file; see section 3.3.1.
2. Edit the following settings in the `[follow-on-pipeline]` section:

```
[follow-on-pipeline]
tile=<pipeline configuration>
timeout=<n>
```

   Where

   - `tile` identifies the product tile that launches automatically when the current pipeline exits. Set `tile` to the base name of the product’s pipeline configuration file (that is, the file name minus the .ini extension). For example, if the product’s pipeline configuration file is mt-canvus-2.0.0.ini, set `tile` to:

   ```
   tile=mt-canvus-2.0.0
   ```

     If you do not want any follow-on product to launch, leave `tile` undefined (null):

   ```
   tile=
   ```

     Note: *Only one product can start automatically! The* `tile` *setting must only specify a single product. It cannot specify a comma-separated list of products.*

   - `timeout` specifies an optional pause, in seconds, before launching the next product. The default is zero seconds so the next product launches immediately and users cannot cancel it.

     If you do specify a pause, a countdown timer and Cancel button are displayed. For example, if you want the next product to launch after 15 seconds, enter:

   ```
   timeout=15
   ```

3. Restart MT Launcher for the pipeline changes to take effect; see section 4.

3.3.4 Configure the product log file

Log entries for an individual product are saved to a *pipeline log file*. Pipeline log entries are generated each time a MultiTaction product such as MT Canvus or MT Showcase runs. You can specify where pipeline logs are saved and you can set custom log file names.

Follow these steps:

1. Edit the custom pipeline configuration file; see section 3.3.1.
2. Edit the following settings in the `[logs]` section:

   ```
   [logs]
   path=<folder>
   basename=<name>
   ```

   Where:

   - `path` specifies the folder where pipeline logs are saved. By default, these are saved below the home folder of the runtime user (Ubuntu) or in the runtime user's profile (Windows). For example, if the user logged on while MT Launcher runs is 'multi' then logs are saved here:

     Ubuntu example

     ```
     path=/home/multi/MultiTaction/launcher/logs
     ```

     Windows examples

     ```
     path=C:/Users/multi/AppData/Local/MultiTaction/launcher/logs
     path=C:\\Users\\multi\\AppData\\Local\\MultiTaction\\launcher\\logs
     ```

     Note: *Windows paths must use / forward slashes or \\ double backslashes. They cannot use \ singe backslashes.*

     For details about default folders for pipeline log files, see section 3.4.2.

   - `basename` specifies the base element of the pipeline log file name, where the full name is <basename>-<date>.log.

     We recommend that `basename` matches the product name. For example, if you have a pipeline for MT Canvus 2.0, use this base element:

     ```
     basename=mt-canvus-2.0
     ```

     When the timestamp is added, this will generate log files like this:
     mt-canvus-2.0-2018-01-22.log

     Likewise, if you have set up a pipeline to launch a specific MT Showcase app (see section 5.1), set `basename` to match the app name:

     ```
     basename=mt-showcase-1.8-my_app
     ```

     If `basename` is undefined (null), it defaults to match the pipeline configuration file name. For example, if the configuration file is mt-canvus-2.0.0.ini, then the pipeline log files default to mt-canvus-2.0.0-<date>.log.

3. Restart MT Launcher for the pipeline changes to take effect; see section 4.

Now edit the process blocks in the custom pipeline configuration file; continue to section 3.3.5.

3.3.5 Edit the process blocks

If you want to configure the launch of individual programs, you must edit the *process blocks* in your custom pipeline configuration files. A process block specifies: which executable to run, the working directory for the process; arguments passed to the executable; restart settings; library paths; the process mode; and so on.

Each pipeline configuration file can include one or more process blocks, each with the settings described below.
Follow these steps:

1. Edit the custom pipeline configuration file; see section 3.3.1.
2. Edit the following settings in the `[process]` section:

```
[process-<n>]
exec=<executable>
working-directory=<path>
arguments=<executable arguments>
restart-on-crash=<true or false>
restart-limit=<n>
restart-period=<n>
required=<true or false>
library-paths=<path list>
invalid-config-exit-codes=<exit codes>
mode=<sync, async or daemon>
environment\<name>=<value>
add-log-timestamp=false<true or false>
```

   See below for settings descriptions.

3. Restart MT Launcher for the pipeline changes to take effect; see section 4.

Where:

- `[process-<n>]` identifies a process configuration block and `<n>` determines the order in which process blocks are executed in the pipeline. For example, a `[process-0]` block is executed first and a `[process-1]` block is executed second.

  Note: *The execution order can differ from the order in which the blocks are listed in the pipeline configuration file.*

- `exec` specifies the executable to run, plus its path (relative or absolute). Relative paths are relative to the working directory of the process (not the working directory of MT Launcher). For example:

  Example paths on Ubuntu systems
  Absolute: `exec=/opt/mt-canvus-<ver>/bin/mt-canvus-app`
  Relative: `exec=mt-canvus-app`

  Example paths on Windows systems
  Absolute: `exec=C:/program Files/MT Canvus/bin/mt-canvus-app.exe`
  Relative: `exec=mt-canvus-app.exe`

  Note: *Windows paths must use / forward slashes or \\ double backslashes. They cannot use \ singe backslashes.*

- `working-directory` specifies the working directory for the process. If the `exec` setting specifies:
    - An *absolute* path, then `working-directory` defaults to the path for the process executable, as specified by `exec`. If this default working directory is acceptable, you can omit `working-directory` from the process block (do not set it to null). Alternatively, you can specify a custom working directory; this will depend on the requirements of the process.
    - A *relative* path, then `working-directory` must be explicitly set. For example, the typical working directory for MT Canvus is:

      Ubuntu systems
      ```
      working-directory=/home/multi/
      ```

      Windows systems
      ```
      working-directory=C:/Program Files/MT Canvus/bin
      ```

      Note: *Windows paths <u>must</u> use / forward slashes or \\ double backslashes. They cannot use \ singe backslashes.*

- `arguments` specifies any executable argument(s). Use a comma-separated list for multiple arguments. For example, the default arguments for MT Showcase are:

```
arguments=--minidump-upload-url,
  http://diagnostics/multitaction.com/crash-reports/upload-crash-dump,
  --cef-ignore-gpu-blacklist
```

  You can also use the `arguments` setting to open a specific MT Showcase app and include an Exit button in the welcome screen; see section 5.

- `restart-on-crash` determines whether to restart the process if it crashes. By default, this setting is `false`.

- `restart-limit` specifies the maximum number of permitted restarts within the restart period (see below) before the process is marked as a failure. By default, 3 restarts are permitted.

- `restart-period` specifies the restart period, in seconds. If the number of restarts during this period exceeds the restart limit (see above), the process is marked as a failure. By default, the restart period is 10 seconds.

- `required` determines whether the process is required by the pipeline. If a required process is marked as a failure (it fails to restart; see above), the entire pipeline is marked as a failure and terminated. By default, this setting is `false`.

- `library-paths` specifies a comma-separated list of library paths for the process.

  These will modify the library path for the process. (PATH on Windows systems; LD_LIBRARY_PATH on Linux systems)

  This setting overwrites any system library path.

- `invalid-config-exit-codes` specifies a comma-separated list of exit codes that are interpreted as invalid configuration settings by the pipeline. The default code for this setting is 100.

- `mode` specifies the *process mode*. This mode determines how the pipeline handles the process. The following options are available:

  `sync` The pipeline waits for the process to finish before starting the next process in the pipeline.

  `async` The pipeline immediately starts the next process without waiting for this process to finish. The pipeline waits for the process to finish before the pipeline terminates.

  `daemon` The pipeline immediately starts the next process without waiting for this process to finish. All `daemon` processes are terminated when there are no `sync` or `async` processes still running.

  The default mode is `daemon`.

- `environment\<name>=<value>` specifies a set of environment variables for the process as *key/value* pairs. The syntax is environment\ then key=value assigment.

  For example: environment\USER=multi

- `add-log-timestamp` determines whether MT Launcher prefixes all lines of output from the process with a timestamp. By default, this setting is `false`.

3.4 Log files

MT Launcher generates its own log files and log files for each *pipeline* (see section 1.1).

3.4.1 MT Launcher log files

Log entries are generated when MT Launcher starts, and each time a MultiTaction product such as MT Canvus or MT Showcase starts and exits.

- **Ubuntu application computers:** Find the log files here:
  ~/MultiTaction/launcher/logs

  Where ~/ refers to the home folder of the *runtime user*. If the user logged on while MT Launcher runs is 'multi', the expanded path is:
  /home/multi/MultiTaction/launcher/logs

- **Windows application computers:** Find the log files here:
  %LOCALAPPDATA%\MultiTaction\launcher\logs

  Where %LOCALAPPDATA% refers to the *runtime user's* profile. If the user logged on while MT Launcher runs is multi, the expanded path is:
  C:\Users\multi\AppData\Local\MultiTaction\launcher\logs

Note: *The* `rotate-period` *setting in launcher.ini defines the maximum age of uncompressed log files. When MT Launcher starts, any log files older than this are compressed to .gz files. See section 3.2.*

3.4.2 Pipeline log files

Log entries for MultiTaction products such as MT Canvus or MT Showcase are saved to a *pipeline log file*. Pipeline log entries are generated each time a product is launched from MT Launcher. Log entries record the individual start-up processes, product activity (that is, conventional log entries for MT Canvus or MT Showcase), and processes when the product exits back to MT Launcher.

You can configure a product's log file name and folder in the product's pipeline configuration file; see section 3.

- **Ubuntu application computers:** By default, pipeline log files are here:
  ~/MultiTaction/launcher/logs

  Where ~/ refers to the home folder of the *runtime user*. If the user logged on while MT Launcher runs is 'multi', the expanded path is:
  /home/multi/MultiTaction/launcher/logs

- **Windows application computers:** By default, pipeline log files are here:
  %LOCALAPPDATA%\MultiTaction\launcher\logs

  Where %LOCALAPPDATA% refers to the *runtime user's* profile. If the user logged on while MT Launcher runs is 'multi', the expanded path is:
  C:\Users\multi\AppData\Local\MultiTaction\launcher\logs

4 Start or exit MT Launcher

MT Launcher is designed to run on video walls and provide end-users with a simple method for *launching* products such as MT Canvus or MT Showcase. It is *not* designed to allow end-users to easily exit to the desktop from MT Launcher.

However, administrators can use a keyboard shortcut to exit from MT Launcher if, for example, they want to access the OS desktop. It is also possible to display an Exit tile in MT Launcher.

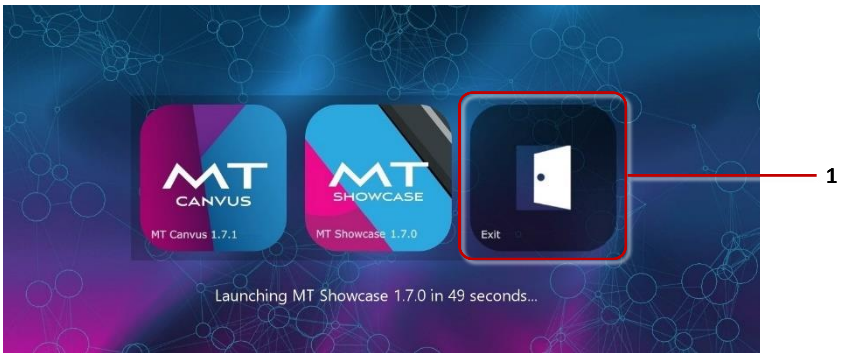
To start MT Launcher:

- **Ubuntu systems:** Do one of the following:
  - Right-click the desktop and click MT Launcher in the menu.
  - Right-click the desktop and launch a terminal emulator. Then run this command:
    ```
    $ mt-launcher
    ```
- **Windows systems:** Double-click the MT Launcher desktop shortcut:

To subsequently exit MT Launcher and access the OS desktop:

- **Ubuntu *or* Windows systems:** Do one of the following:
  - Click MT Launcher and press Ctrl+Q.
  - Tap the Exit tile (if enabled).

    Note: *If you installed MT Launcher as part of the MultiTaction base image, the Exit tile is disabled by default. To enable it, you must edit the* `allow-exit` *setting in the custom version of launcher.ini; see section 3.2*



Launching MT Showcase 1.7.0 in 49 seconds...

Example MT Launcher menu with Exit tile (1). This tile is for administrators and allows them to quickly exit MT Launcher and access the OS desktop.

5 MT Showcase customizations

You can edit the pipeline configuration file for MT Showcase to launch a specific app or display the Welcome screen. You can also display an Exit button on the welcome screen.

5.1 Launch a specific MT Showcase app

Note: *This feature uses the* `--app` *argument. This argument is only supported in MT Showcase 1.7 or later.*

By default, when you use MT Launcher to launch MT Showcase, it automatically opens the most recent app ie, the app that was running in the previous session. But you can override this behavior, customizing the MT Launcher tile to launch a specific MT Showcase app from MT Launcher.

Follow these steps:

1. Edit the *custom version* of the pipeline configuration file for MT Showcase. For details about where to find this file, see section 3.1.3.

2. Go to the `[ui]` block and set the `name` setting to reflect the name of your MT Showcase app. This name appears on the MT Launcher tile. For example, if your app is named Showcase Tutorial:

   ```
   [ui]
   name=Showcase Tutorial
   ```

   You do not need to enclose the app name in quotes if it contains spaces.

3. Go to the MT Showcase process block and edit the `arguments` setting. You must append `--app,<app name>` to the default list of arguments:

   ```
   [process-<n>]
   arguments=<default arguments>,--app,<app name>
   ```

   For example, if the MT Showcase process block is sequenced to execute first and your app is named Showcase Tutorial, the full argument list, including the default arguments, looks like this:

   ```
   [process-0]
   arguments=--minidump-upload-url,
     http://diagnostics/multitaction.com/crash-reports/upload-crash-dump,
     --cef-ignore-gpu-blacklist,
     --app,Showcase Tutorial
   ```

   Note: *The arguments setting is described in section 3.3.5.*

4. Restart MT Launcher.

Note: *An app is an individual MT Showcase application. An app defines the actual content available to users on the screen (images, videos, PDFs, and so on), plus the appearance and behavior of screen items such as finger menus and the background. For more about apps, see the latest MT Showcase Editor Manual; registered users can download this manual from* https://cornerstone.multitouch.fi/mt-showcase-manuals.

5.1.1 If the MT Showcase app no longer exists

If you set up an MT Launcher tile to launch a specific MT Showcase app, you need to keep the tile and app in sync. If the app is deleted or renamed, you need to delete or reconfigure the tile.

But mistakes happen! If a user taps an MT Launcher tile to launch an MT Showcase app that no longer exists, the user is taken instead to the MT Showcase welcome screen. For this reason, you may want to add an Exit button to the MT Showcase welcome screen, allowing users in this situation to quit back to MT Launcher; see section 5.3.

5.2 Launch the MT Showcase welcome screen

Note: *This feature uses the* `--app` *argument. This argument is only supported in MT Showcase 1.7 or later.*

You typically use MT Launcher to launch an MT Showcase app, but you can set up a tile to open the MT Showcase welcome screen. Follow these steps:

1. Edit the *custom* version of the pipeline configuration file for MT Showcase. For details about where to find this file, see section 3.1.3.

2. Go to the `[ui]` block and set the `name` setting to identify the MT Showcase welcome screen. This name appears on the MT Launcher tile. For example:

   ```
   [ui]
   name=Showcase 1.7 Welcome
   ```

   You do not need to enclose the name in quotes if it contains spaces.

3. Go to the MT Showcase process block and edit the `arguments` setting. You must append `--app,none` to the default list of arguments:

   ```
   [process-<n>]
   arguments=<default arguments>,--app,none
   ```

   Note: *The* `arguments` *setting is described in section 3.3.5*.

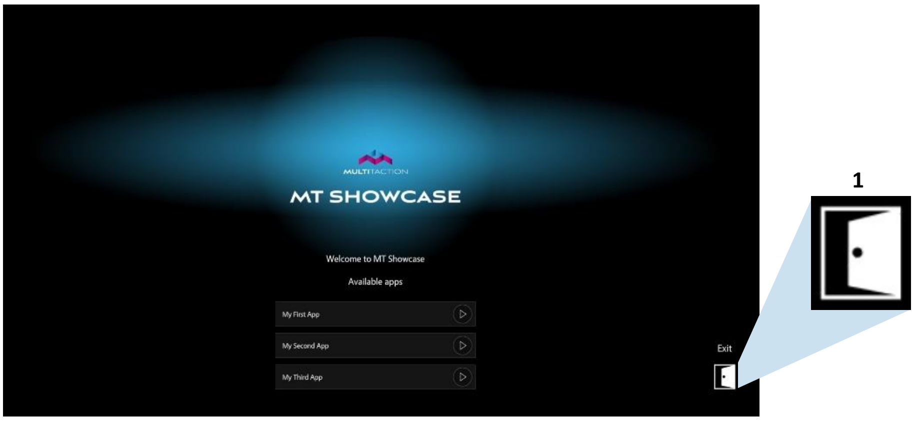
4. Restart MT Launcher.

You can also display an Exit button on the welcome screen; see section 5.3.

5.3 Display an Exit button on the MT Showcase welcome screen

Note: *This feature uses the* `--show-exit` *argument. This argument is only supported in MT Showcase 1.7 or later.*

You can display an Exit button on the MT Showcase display welcome screen, allowing users to quit back to MT Launcher. This can be useful if users cannot run the MT Showcase app they want (for an example of how this can happen, see section 5.1.1).



MT Showcase welcome screen with (**1**) *Exit button*

To display the Exit button, follow these steps:

1. Edit the *custom version* of the pipeline configuration file for MT Showcase. For details about where to find this file, see section 3.1.3.

2. Go to the MT Showcase process block and edit the `arguments` setting. You must append `--show-exit` to the default list of arguments.

```
[process-<n>]
arguments=<default arguments>,--show-exit
```

   For example, if the MT Showcase process block is sequenced to execute first and your app is named Showcase Tutorial, the full argument list, including the default arguments and `--show-exit`, looks like this:

```
[process-0]
arguments=--minidump-upload-url,
  http://diagnostics/multitaction.com/crash-reports/upload-crash-dump,
  --cef-ignore-gpu-blacklist,
  --show-exit,
  --app,Showcase Tutorial
```

   Note: *The* `arguments` *setting is described in section 3.3.5*.

3. Restart MT Launcher.

Important! *Do not display the Exit button if MT Showcase starts from a launch script (mt-showcase.sh or mt-showcase.bat). Here, the Exit button will simply return the user to the launch script, which will rerun automatically and restart MT Showcase!*

6 MT Canvus customizations

You need to edit the pipeline configuration file for MT Canvus if your *working version* of mt-canvus.ini is installed to a custom location.

Follow these steps:

1. Edit the *custom version* of the pipeline configuration file for MT Canvus. For details about where to find this file, see section 3.1.3.
2. Go to the MT Canvus process block and edit the `arguments` setting. You must append `--mt-canvus-config,<file location>` to the default list of arguments:

   ```
   [process-<n>]
   arguments=<default arguments>,-- mt-canvus-config,<file location>
   ```

   Where `<file location>` specifies the full path to the working version of mt-canvas.ini.

   For example, if the MT Canvus process block is sequenced to execute first, the argument list looks like this:

   ```
   [process-0]
   arguments=<default arguments>,
     --mt-canvus-config,/home/custom/mt-canvus.ini
   ```

3. Restart MT Launcher.