

Files and directories of the Pixet and SDK

Online version:

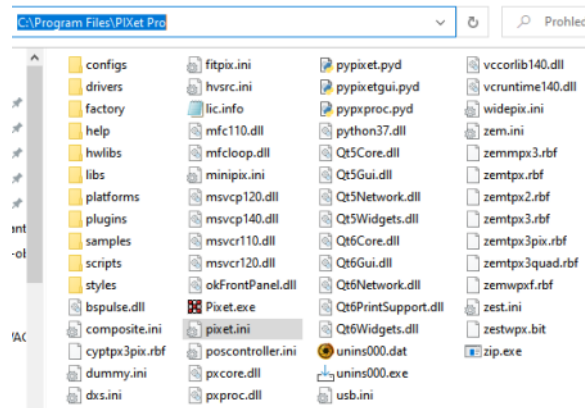
https://wiki.advacam.cz/wiki/Files_and_directories_of_the_Pixet_and_SDK

Contents

Main directory of the Pixet program	3
User data directory	3
Subdirectories of the Pixet program	4
<i>The factory and the configs subdirectories</i>	4
factory	4
configs	4
logs	5
hwlibs	5
plugins	5
drivers	5
libs	5
scripts	6
Other subdirs	6
Main directory of the API-using programs, independent on the Pixet	6
Special files	8
Configuration XML files	8
pixet.ini file	8
[Settings]	8
[Hwlibs]	10
[Plugins]	10
[PixelmanPlugins]	10
Device configuration ini files	10
Device firmware files	11
Insidious complications with files/dirs	11
Related	12



Main directory of the Pixet program



Pixet main directory if all here

Here are main program files, binary libraries, API files, licence file and the pixet.ini file. Here may also be hwlibs subdirectory, special files for some devices. Their location depends on the [Hwlibs] section of the pixet.ini file. There can also be subdirectories factory, configs and logs. They are there if UseAppDataDir is disabled in pixet.ini, in the [Settings] section. If here are not located configs, inis and logs, this directory can be located in Program files, or in another location where the user cannot normally write.

User data directory

This is location of [configuration](#) and [log](#) files. The program must be able to write here. This is used, if in the [pixet.ini](#), in the [Settings] section is located UseAppDataDir=true (default state).

AppData - user data directory

Default location is in the user's system application data directory, like us:

- C:\Users\username\AppData\Local\PixetPro on Windows
- ~/.config/PixetPro on Linux
- ~/Library/Application Support/PixetPro on MacOS

Notes

- Location of the directory can be changed anywhere, using the AppDataDir= line in the pixet.ini.
- Turning on UseAppDataDir does not affect the factory directory. If it is not changed using factoryDir=, it is still expected beside the running program, or in the current directory, if different.

See: [#pixet.ini file](#)

Subdirectories of the Pixet program

The factory and the configs subdirectories

The factory directory should contain the factory default configuration XML files. The Pixet core use it while starting, if the configuration file is not in the configs directory. The configs directory contain configuration XML files. The Pixet core try to use it while starting and save the current settings to it, if exiting.

(This process works the same way when you start and quit the Pixet program or if you use C API functions `pxclnit/pxcExit` or Python API functions `pypixet.start()` / `pixet.exitPixet()`+`pypixet.exit()`.)

The detector cannot work properly if proper config not loaded.

See [#Configuration XML files](#)

Location of this subdirs can be in the main program directory or in user data directory, depending on the `UseAppDataDir=` line in the `pixet.ini` file.

factory

It should contain the factory default configuration XML files. The Pixet core use it while starting, if the configuration file is not in the configs directory or if user click the "Load factory config" or scripts can use it by the `loadFactoryConfig()` method. Location of this subdir can be in the main program directory or can be located anywhere using the `FactoryDir=` line in the `pixet.ini` file.

Hint: If you frequently do experiments with more instances of Pixet, API-using programs or scripts and have more devices, you can use single location of the factory directory for all.

configs

Contain configuration XML files for devices. The Pixet core try to use it while starting the Pixet program or in `pypixet.start()` method in scripts, or if the C API function `pxclnit()` used. And automatically save the current settings to it, while exiting the Pixet program, or if the `pypixet.exit()` method is used in scripts, or if the `pxcExit()` function used via C API.

In addition, this directory contains:

- Configuration XML files for filters - name format `filtername_DevName ChipID.xml` - for example "adaptiveff_MiniPIX D06-W0065.xml"
- Configuration INI files for plugins - dev specific - name format `pluginname_ChipID.ini` - for example "devcontrol_D06-W0065.ini"
- Configuration INI files for plugins - name format `pluginname.ini` - for example "pyscripting.ini"

Location of this subdir can be in the main program directory or in user data directory, depending on the `UseAppDataDir=` line in the `pixet.ini` file, or can be located anywhere using the `ConfigsDir=` line in the `pixet.ini` file.

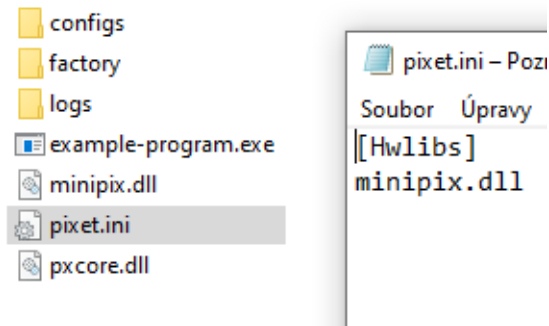
logs

The logs directory

Here are log files from all hardware libraries, pixet core. Files are from last Pixet core run. While next run, the core archive last log files. Subdirectories logs1, logs2, ... contains backups for last 10 sessions.

Location of this subdir can be in the main program directory or in user data directory, depending on the UseAppDataDir= line in the pixet.ini file, or can be located anywhere using the LogsDir= line in the pixet.ini file.

hwlibs



Directory of the minimalistic program using Advacam C API, working on computer with .NET installed

This is the usual place for hwlib libraries, designed to work with different types of devices. The actual location of the files is defined in the [hwlibs] section of the pixet.ini file. Default is the hwlibs subdirectory of the main program running directory. But here are a simple relative paths from the main program location and can be configured without subdirectory. It is commonly used in single-purpose programs using API.

See [#pixet.ini file](#)

plugins

Default location of plugins to the Pixet program.

Warning: Some plugins can be need for API too.

drivers

Installators of drivers for some devices.

libs

Libraries for Python scripting plugin.

scripts

- Sample scripts for using Python scripting plugin.
- The default place to search for imports used in scripts run from the Python scripting plugin.

Other subdirs

- samples - Sample data for demo when using virtual file device.
- help - PDF manuals
- platforms, styles - System directories of QT.

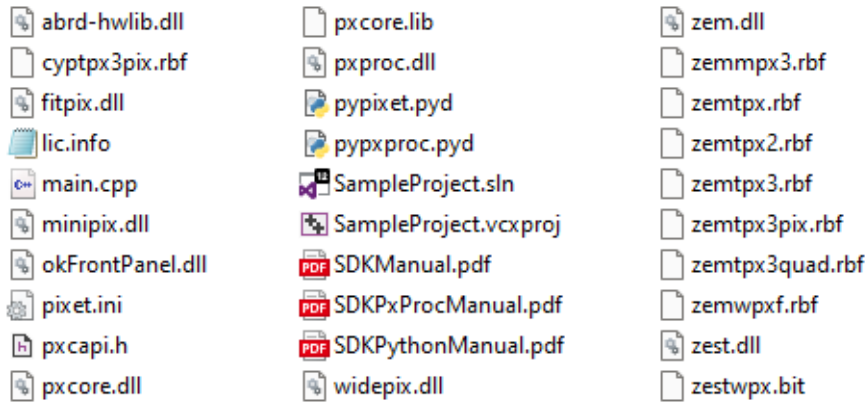
Main directory of the API-using programs, independent on the Pixet

User can write programs using our API. To operate them, need:

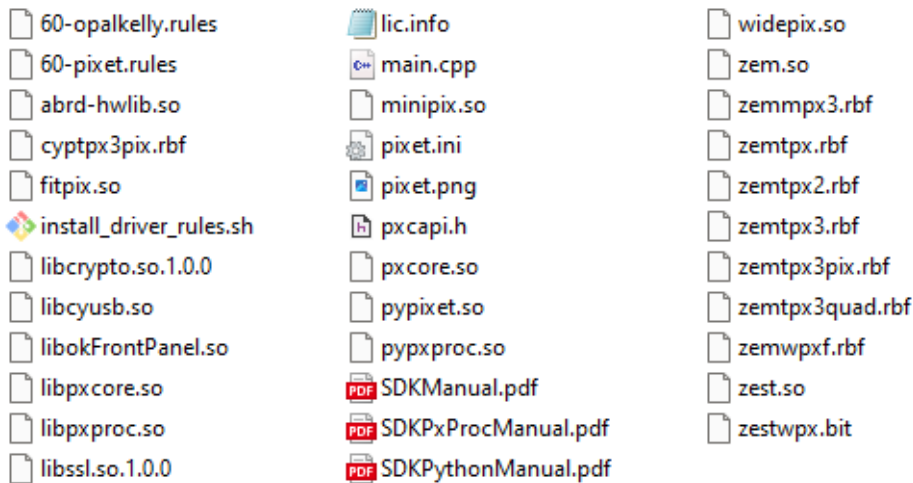
- Pixet core library - The core of the Pixet program. pxcore.dll, or pxcore.so file.
- Pixel processing library - pxproc.dll or pxproc.so library - only for Clustering and Spectral imaging use
- Static linking libraries - To compile binary programs on Windows, user need the pxcore.lib (and pxproc.lib). Only in compile time. Compiler must be setup for using it.
- Auxilliary libraries of system - If program must working on computer without an IDE installed, can need some auxilliary libs, like as: msvcp140.dll, vcruntime140.dll, etc
- pixet.ini - configuration for the Pixet core. See [#pixet.ini file](#)
- HWlibs for the given hardware (and firmwares if devices need). See [#hwlibs](#) and [#Device firmware files](#)
- Device configuration files in the factory and configs directories, or elsewhere according to the settings in pixet.ini. See [#The factory and the configs subdirectories](#)
- Place for logs in the logs directory, or elsewhere according to the settings in pixet.ini. See [#logs](#)
- Python interface - To run Python scripts, pypixet.pyd is needed in addition. And a pypxproc.pyd for Clustering and Spectral imaging objects.

Basic files are in the API packages:

The Pixet API packages



Files in the Pixet API package for Windows



Files in Pixet API package for Linux

Download at advacam.com/downloads/

Note: Pay attention to download the correct environment version (OS, 32/64 bit, x86/ARM)

pxcore.dll (or lib...so)	Binary libraries (pxcore.lib is only for Windows binary compile)
pxcore.lib	
pxproc.dll (or lib...so)	
pypixet.pyd (or .so)	Python core and processing libraries
pypxproc.pyd (or .so)	See Python API
pxcapi.h	Header for the binary compile with pxcore.dll
	See Binary core API
pixet.ini	Pixet core configuration file
	See: Files and directories of the Pixet and SDK - pixet.ini file
main.cpp	Sample Visual studio project
SampleProject.vcxproj	(Windows package only)
SampleProject.sln	
*.rbf *.bit	Firmware files
install_driver_rules.sh	Linux drivers installer and it's helper files
60-opalkelly.rules	
60-pixet.rules	
okFrontPanel.dll	Helper library of the zem.dll/so hwlib
libokFrontPanel.so	
	Hwlib files (can be in separate directory)
other files .dll (or .so - no lib...so)	See: Files and directories of the Pixet and SDK - hwlibs
lic.info	License file for the Pixet core - not important in this version, but may be in future
pdf files	Binary and python APIs manuals

Pixet core on Windows need more Microsoft Visual Studio .NET standard dlls (vccorlib140.dll etc).

Alternatives to run the API using programs:

1. Copy all need files to the Project directory.
2. While running the program, change the operating directory to the Pixet main directory.
3. Copy the binary executable or python script to the Pixet main directory and run it from here. Useful for occasional single test, but can do problems in more extensive cases.

Special files

Configuration XML files

The config file contains all detector settings and you can start and continue with the settings as it was while last program exit. The factory config is config saved after low-level detector setup, finding of bad pixels, calibrations and other factory tests.

Notes:

- The detector cannot work properly if proper config not loaded.
- The files must have proper names. "WidePIX-B04-W0535-10.xml" for example. If not, automatic load at start will not work. To obtain proper config name, you can do the program exit and see saved filename in the configs directory.

See:

- [The factory and the configs subdirectories](#)

pixet.ini file

This is the main configuration file of the Pixet core. It has several sections beginning with a title in square brackets. Sections contain configuration lines formatted as Name=value or Name= for default value. They can also contain comments, which start with a semicolon. This way you can also deactivate some configuration lines. Default version of the file containing used options and all remaining options disabled + with =default. User can edit it and enable.

Warning:

Up to release 1.8.3 the [Hwlibs] section contains lines without the = symbol. This is not compliant with the INI standard. The file format will change soon. From the next version all data lines will start with the name=.

The default path where Pixet core looks for the pixet.ini file is the same as the path to the program that started the pxcore library. So the file should be next to it. Attention, all other paths are relative to the current directory, not to the position of the program.

So if you run c:\program files\pixet\pixet.exe from c:\users\myname, pxcore will try to load c:\program files\pixet\pixet.ini and all other relative paths (including those listed in the pixet.ini) will start with c:\users\myname.

[Settings]

General settings of used directories, core and the Pixet program parameters:

- UseAppDataDir UseAppDataDir=true - Change default location of the configs and logs subdirs and ini files to the user data directory. It also allows different login users to have different settings of same devices in same programs.

Note: This does not affect the factory directory.

See: [User data directory](#)

- `PixetIniInAppDataDir PixetIniInAppDataDir=true` - Enable reading the `pixet.ini` file from the user data directory. Very useful if the program is located in the Program files or at other location with no write rights. It also allows different users to have different `pixet.ini`. Note: Enabling this option means that the Pixet core first loads this file (from the program launch point) and, after detecting this option, loads it again from the user data directory. Other settings in the default `pixet.ini` are therefore ignored.

See: [User data directory](#)

- `PixetIni PixetIni=C:\myPixetAppData\myPixetIni1.ini` - Change name and location of the `pixet.ini` file. Similar to the case of `PixetIniInAppDataDir`, when using `PixetIni=`, the initialization file will be loaded again and the contents of the original file will not affect the further operation of the software.
- `AppDataDir AppDataDir=C:\myPixetAppData\group1` - Change location of the user data directory to specified path. Useful if user have more programs using a Advacam devices and want divide it to some groups with different settings.
- `ConfigsDir ConfigsDir=C:\myPixetAppData\configsGroup1` - Change location of the configs dir to other location. Useful if user have more programs using a Advacam devices and want divide it to some groups with different settings of devices.
- `FactoryDir FactoryDir=C:\myPixetAppData\factory` - Change location of the factory dir to other location. Useful if user have many independent places of a programs, creating new programs and testing new devices. User can simply store all factory files in one location. Or it is possible to have such a directory on the company server, where users do not have the right to write.
- `LogsDir LogsDir=C:\temp\pxCoreDebug` - Change location of the logs directory. Useful if user have many independent places of a programs, typically if creating new programs. User can simply collect all logs to one directory so that he does not have to open another one every time.
- `Splash Splash=true` - Use a splash screen when the Pixet program starting. It does not affect programs using the API.
- `FileDevice FileDevice=true` - Enable virtual "file device" independent if physical devices connected.

See: [File device](#)

- `FileDeviceWhenNoDevs = FileDeviceWhenNoDevs=false` - Disable virtual "file device" if no physical devices connected.

See: [File device](#)

- `SaveSettings SaveSettings=false` - Disable automatic save device settings on exit pixet core. It affects Pixet as well as programs that use the API.
- `MaxFrameMemory MaxFrameMemory=1024` - Change the frame memory limit for multiframe acquisitions. Reserved for future use, currently not used.
- `MainUi MainUi=devcontrol` - Change main user interface plugin of the Pixet program. User can programm your own plugins and one can be used instead of the Pixet main window.
- `LogLeaveOldLogs LogLeaveOldLogs=false` - Disable archiving old log files sets.
- `LogRotateCount LogRotateCount=15` - Change number of archived log directories.
- `HighDPI HighDPI=true` - Enable `AA_EnableHighDpiScaling` feature in QT GUI environment. Disable can work only in very old versions of Pixet. Currently is deprecated and allways on.

[Hwlibs]

List of hardware library files/paths used in startup to search and connect a devices. See the hwlibs subdirectory description for details.

Examples: Left is default Pixet configuration with hwlib files in the hwlibs subdirectory, right is simple arrangement with files in the program directory. Last line is disabled in both versions.

[Hwlibs]	[Hwlibs]
hwlibs\minipix.dll	minipix.dll
hwlibs\widepix.dll	widepix.dll
hwlibs\zem.dll	zem.dll
;hwlibs\zest.dll	;zest.dll

User can disable some hwlibs using the ; symbol.

List of HWLIB files: Windows / Linux

- Minipix: minipix.dll / minipix.so
- Widepix with Eth: zest.dll / zest.so
- Widepix with USB2: widepix.dll / widepix.so
- Advapix or Widepix with USB3: zem.dll (and aux. file okFrontPanel.dll - allways in the program directory) / zem.so + okFrontPanel.so
- Advapix new: abrd-hwlib.dll / abrd-hwlib.so
- And customer-specific files for motor driving, x-ray sources, etc.

[Plugins]

List of Pixet plugins files/paths. All files is by default in the plugins subdirectory.

Note

Some plugins are not only for GUI Pixet
Accessing specific files like as HDF5, TIFF
These plugins are also needed when using an API.

[PixelmanPlugins]

List of Pixet plugins files/paths with backward compatibility for old Pixelman program.

Warning:

This is very obsolete. Support for Pixelman related stuff (this ini section, file formats, etc) will be removed soon without replacement.

Device configuration ini files

Some devices can has configuration file. Theese files are located in the program main directory or in the user data directory if the UseAppDataDir=true option used. Files have names identical to the names of the corresponding HWlibs, ending with

.ini. For example, the Ethernet Widepix uses HWlib zest.dll (or zest.so on Linux) and the configuration is stored in zest.ini.

There is usually no need to edit these files, you can find details about them in the device's manual.

Only for network devices it is sometimes useful to define IP addresses on which Pixet core should search for devices. Either because the general search does not work on the given network, or so that the program does not "steal" a device in another room. For example, in zest.ini, each device can have its own section and an IP address defined in it:

```
[Device0]
IP=10.10.1.30
```

```
[Device1]
IP=10.10.1.31
```

Device firmware files

Some devices need firmware files to upload while initialization process. These files are always located in the program directory. Names are typically composed from HWlib name and readout chip name (zemtpx3.rbf for example).

List of this kind of files:

zestwpx.bit, zemwpxf.rbf, zemtpx3.rbf, zemtpx3quad.rbf, zemtpx.rbf, cyptpx3pix.rbf

Note: The standard Pixet distribution that comes with your device only contains the files for that device, so you usually won't see all of these files. And Minipix, for example, doesn't need any of it.

Insidious complications with files/dirs

Let's have the following program and with it pypixet.pyd and other necessary files:

```
import os
os.chdir("test-output")
import pypixet

print("pixet core init...")
pypixet.start()
pixet=pypixet.pixet
devices = pixet.devices()
dev = devices[0]

print("dev.doSimpleAcquisition (3 frames @ 1 sec) - start")
rc = dev.doSimpleAcquisition(3, 1, pixet.PX_FTYPE_AUTODETECT, "example.png")
print("dev.doSimpleAcquisition - end:", rc, "(0 is OK)")

pixet.exitPixet()
pypixet.exit()
```

This program normally works with e.g. Minipix, but does not work with e.g. Advapix. Crashes at `pypixet.start()` or has not any detected devices.

The problem is that hwlibs load their files eg firmware from the current directory. Minipix not need any other files, Advapix need a firmware .rbf file. Moving `os.chdir` after `pypixet.start()` solves the problem.

Related

- [Pixet SDK overview](#)
- [The PIXet program](#)
- [File types](#)