

HighFinesse  
The Standard of Accuracy

HighFinesse Tutorial

Control the wavemeter with your  
own application via the network

September 14, 2022

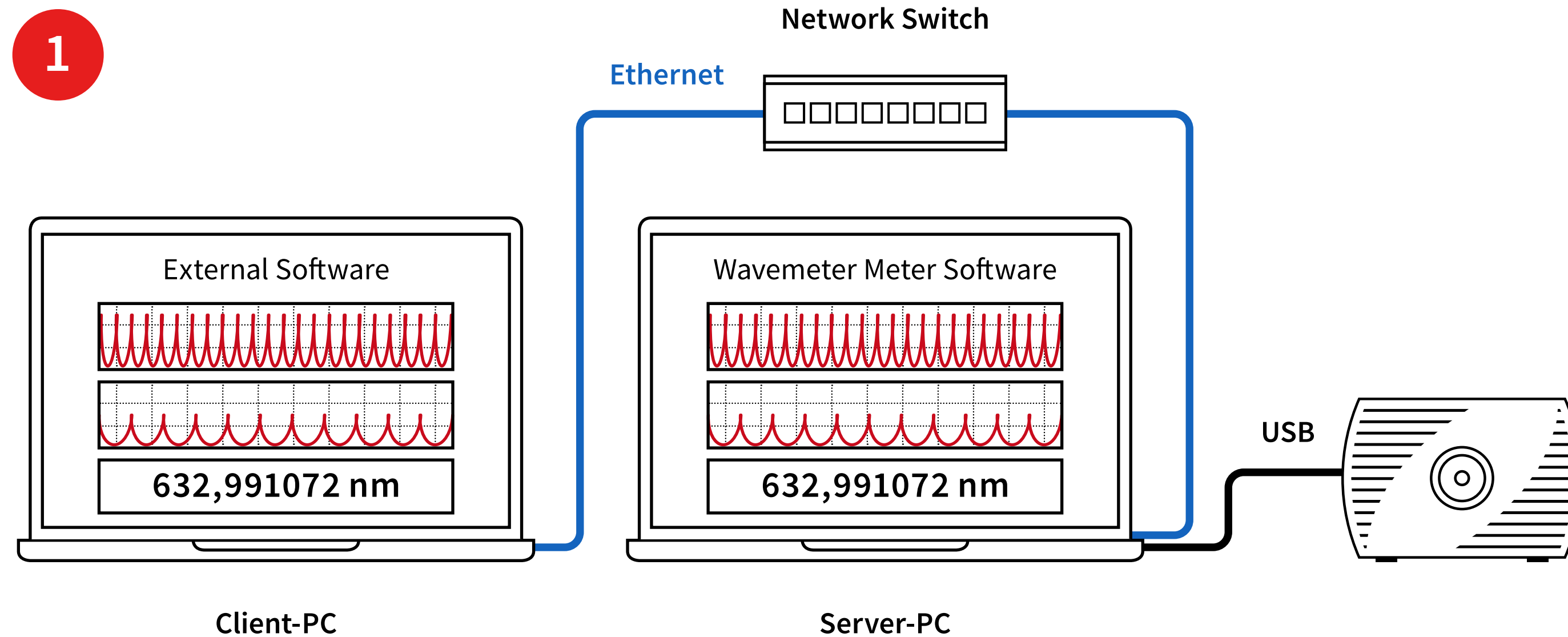
This tutorial shows you how to ...

... control the wavemeter with your own application via the network.

This guide is intended to give you a short introduction on how to control a HighFinesse non-standalone wavemeter or laser spectrum analyzer. It is discussed how to use the LongTerm application and Python example that can be used as a starting point for your own application controlling the wavemeter via the network.

HighFinesse User Manual  
WLM Network Solution



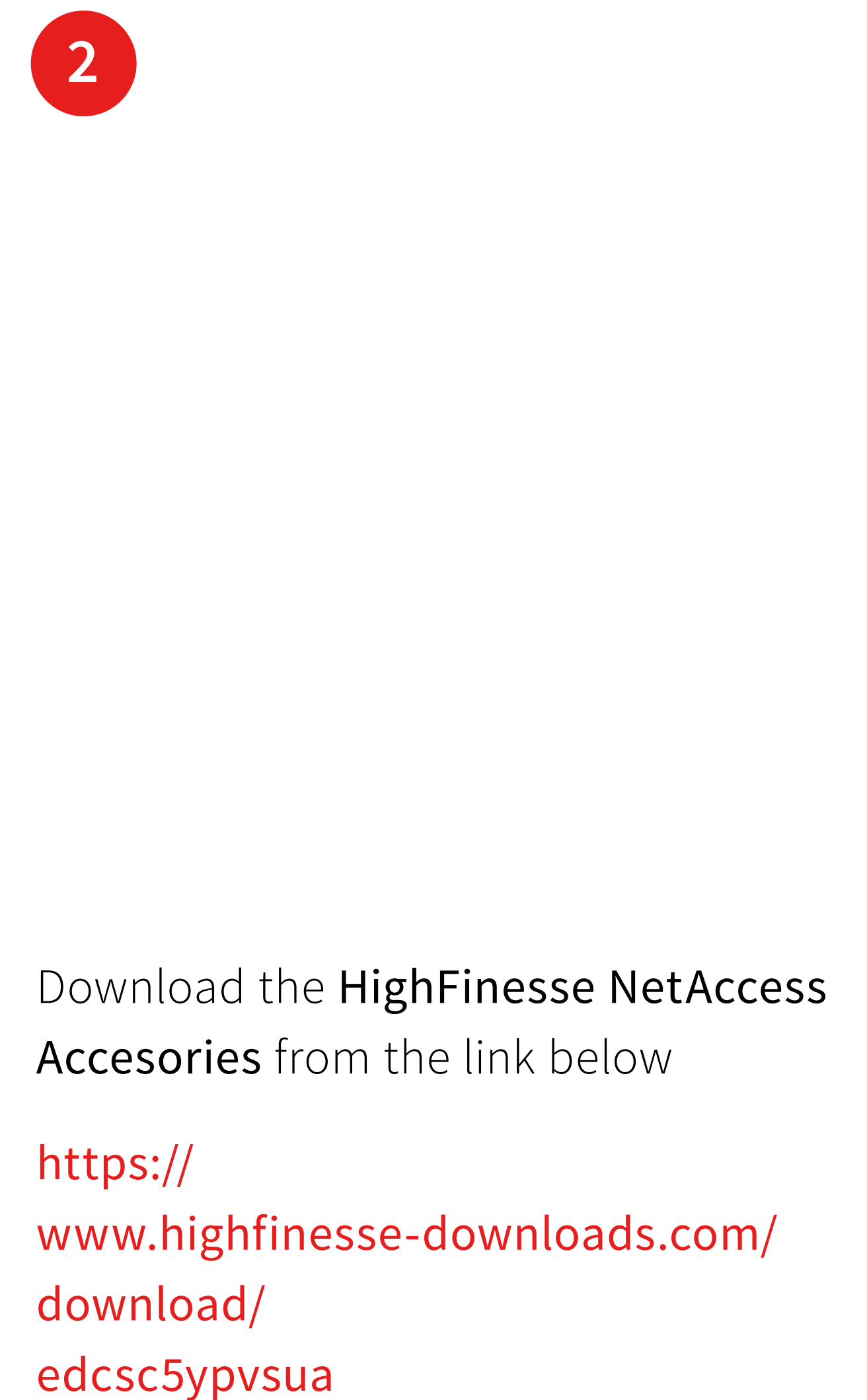


Connect the wavemete to a computer in the network.

Make sure the proper version of Visual C++ Redistributable for Visual Studio for your operating system is installed. You can download it here:

<https://support.microsoft.com/en-us/help/2977003/the-latest-supported-visual-c-downloads>

Then please run the `vc_redist.x64.exe` (`vc_redist.x86.exe` for 32 bit) on both the Server and Client PC.



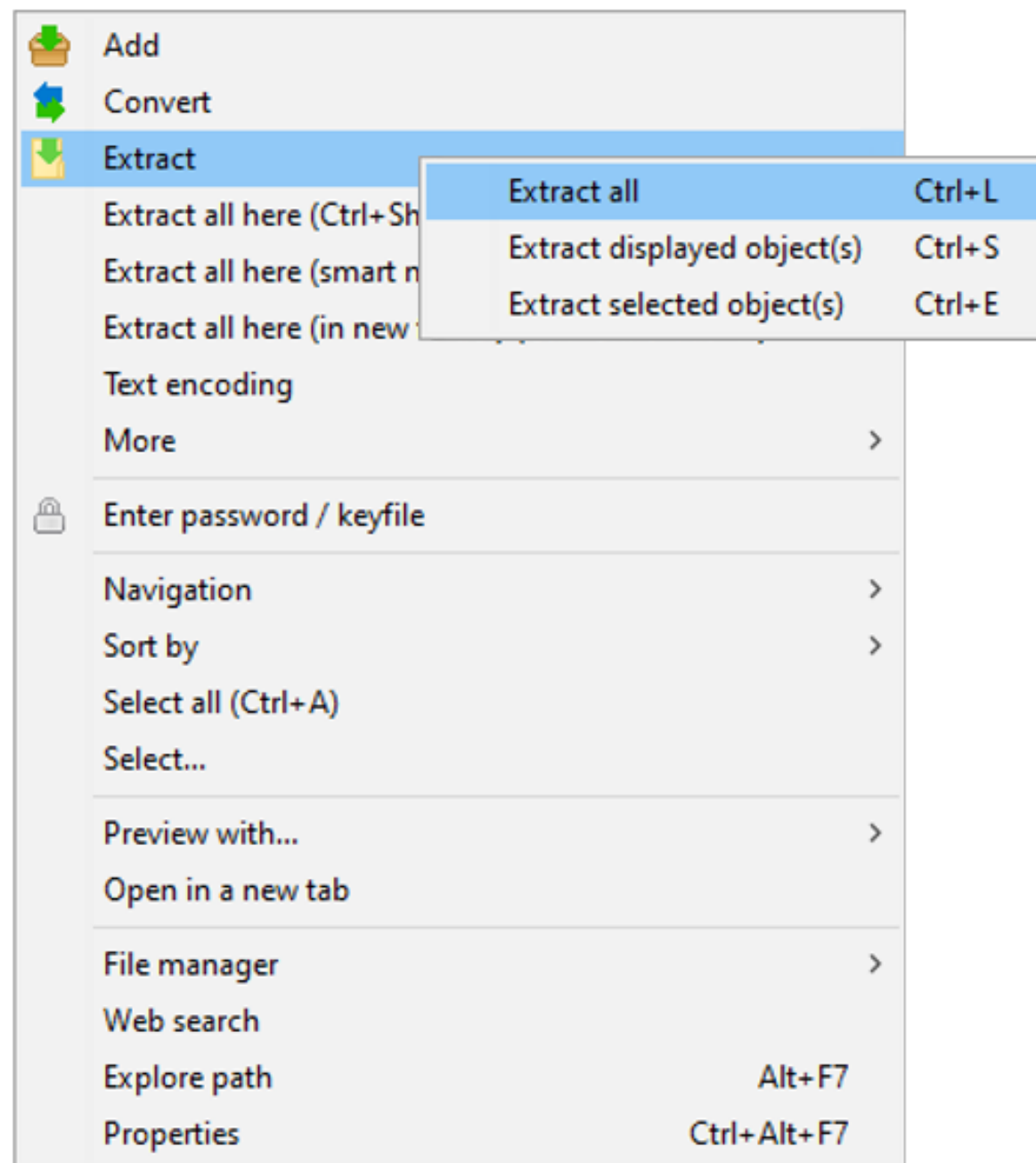
3



HighFinesse NetAccess Accessories.zip  
Type: ZIP archive

Find the ZIP archive  
**HighFinesse**  
**NetAccess Accessories**  
on the **USB stick**.

4



Extract the ZIP archive  
**HighFinesse**  
**NetAccess Accessories**.

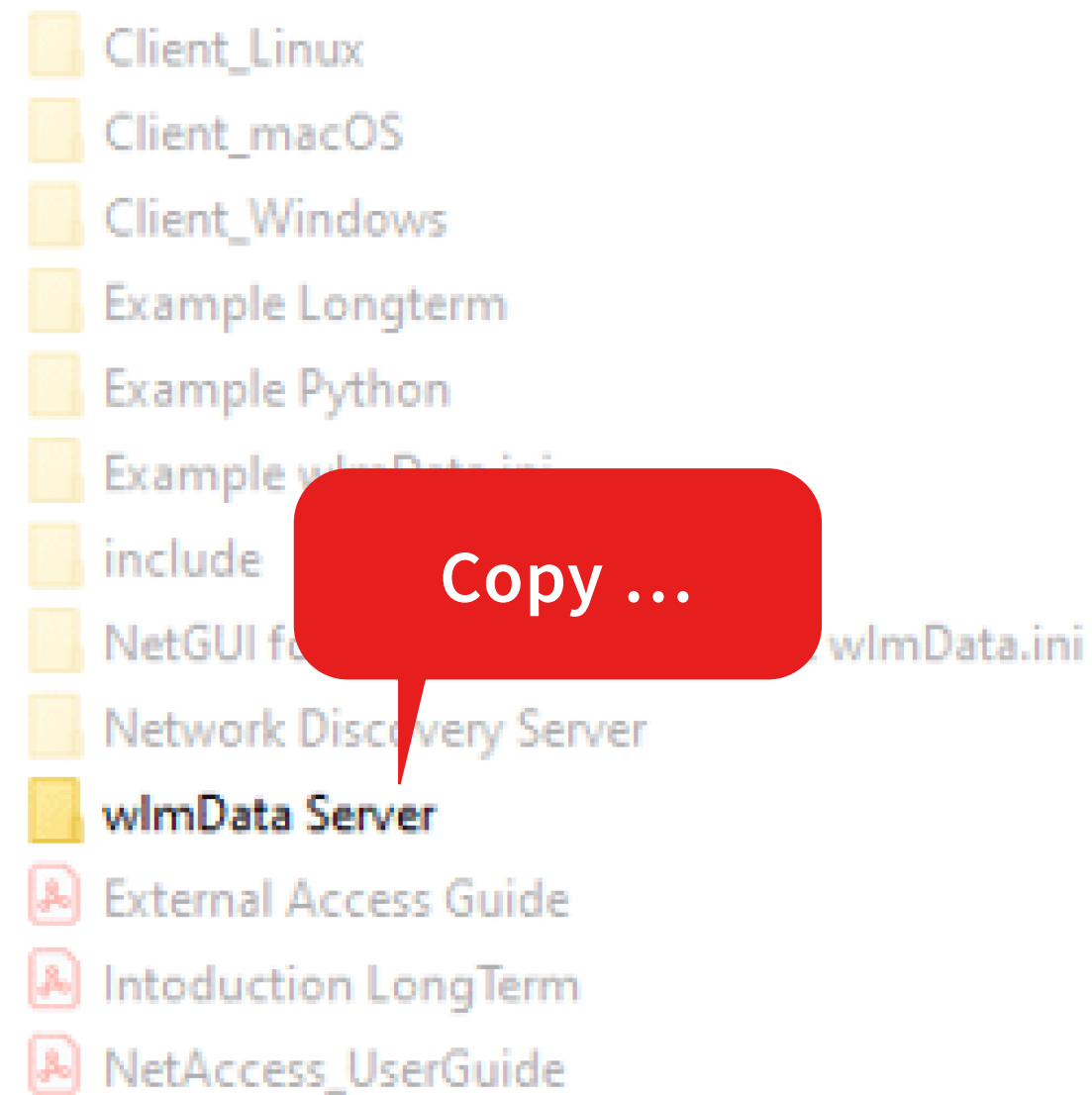
5



HighFinesse NetAccess Accessories  
Type: Folder

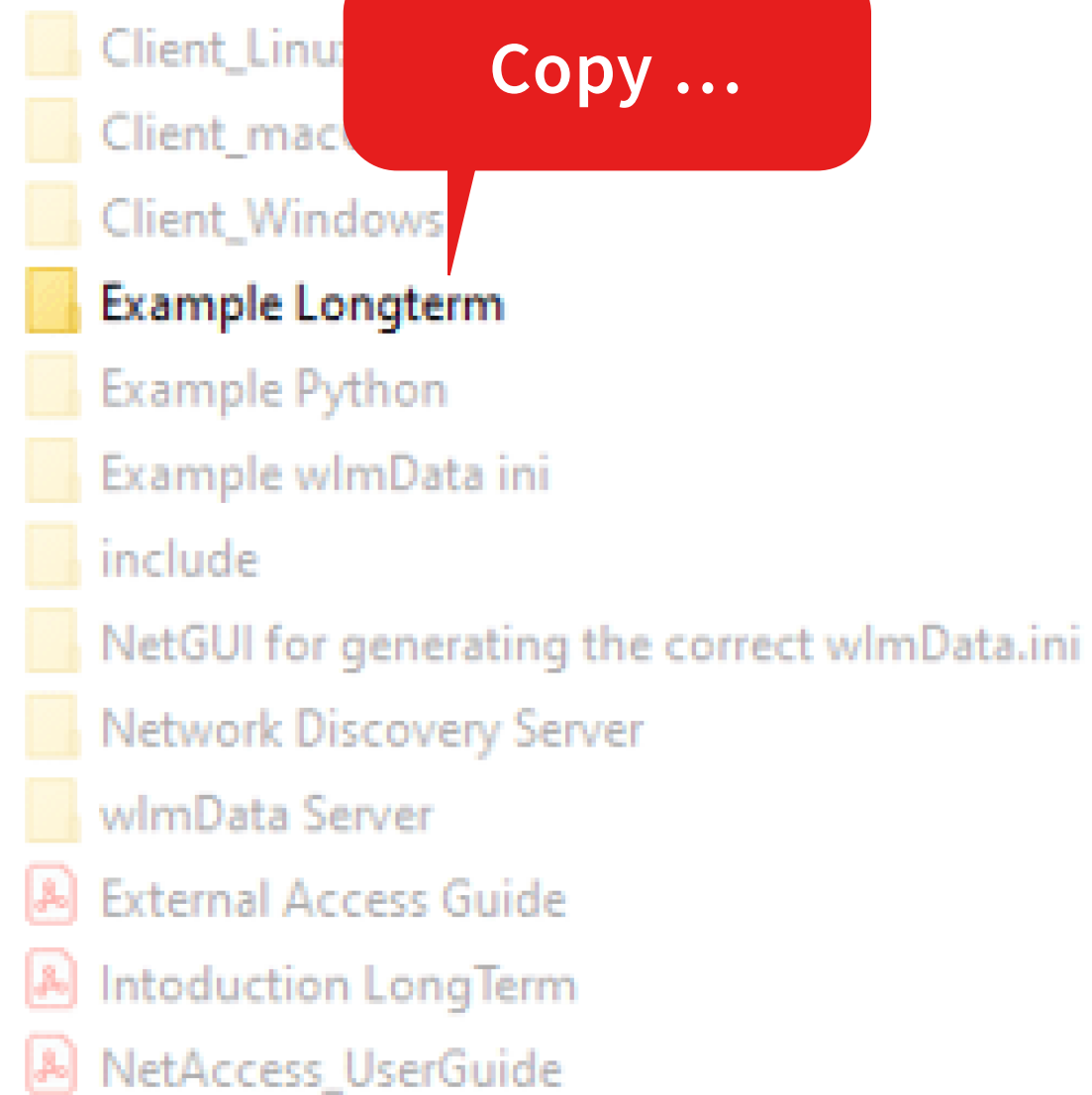
Open the extracted folder  
**HighFinesse**  
**NetAccess Accessories**.

6



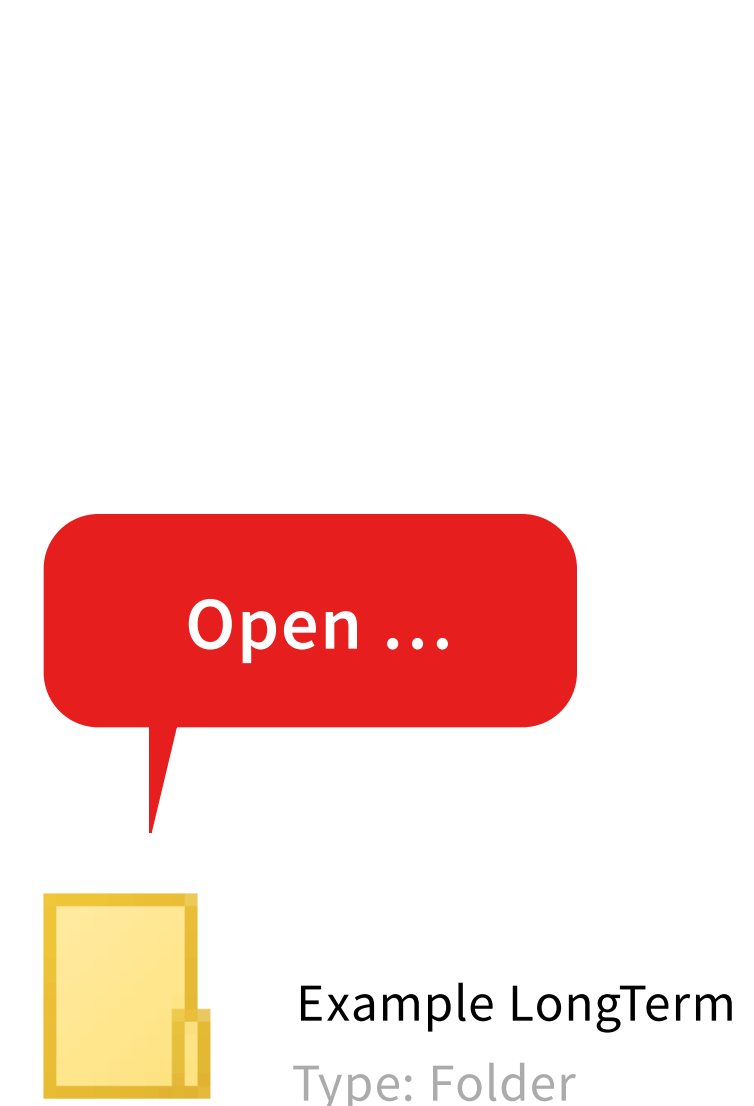
Copy the folder **wlmData Server** to your Server-PC and start the **wlmDataServer.exe**.

7



Copy the folder **Example Longterm** to your client-PC.

8



Open the folder **Example LongTerm ...**

9



Open ... with texteditor

... and open the file wlmData.ini with an texteditor.



10

```

wlmData - Notepad
File Edit Format View Help
; #####
; @file wlmData.ini
; @brief Example configuration file for HighFinesse N
; client side library
; @date: 2021.03.10
; @version: 0.1
; #####

; wlmData.ini example scenario 1 configuration file
; #####

[default]                ; Default settings (i
version = 4                ; IPv4
address = 192.168.13.183   ; Instrument server I
port = 7171                ; Set/Get TCP Port nu
port2 = 7172               ; CallbackProc/Ex TCP
offload = 1                ; ConvertUnit / Conve
                                ; network offload (1=
loglevel = 3               ; Loglevel: Warning
errormode = 9              ; Error signaling: Lo

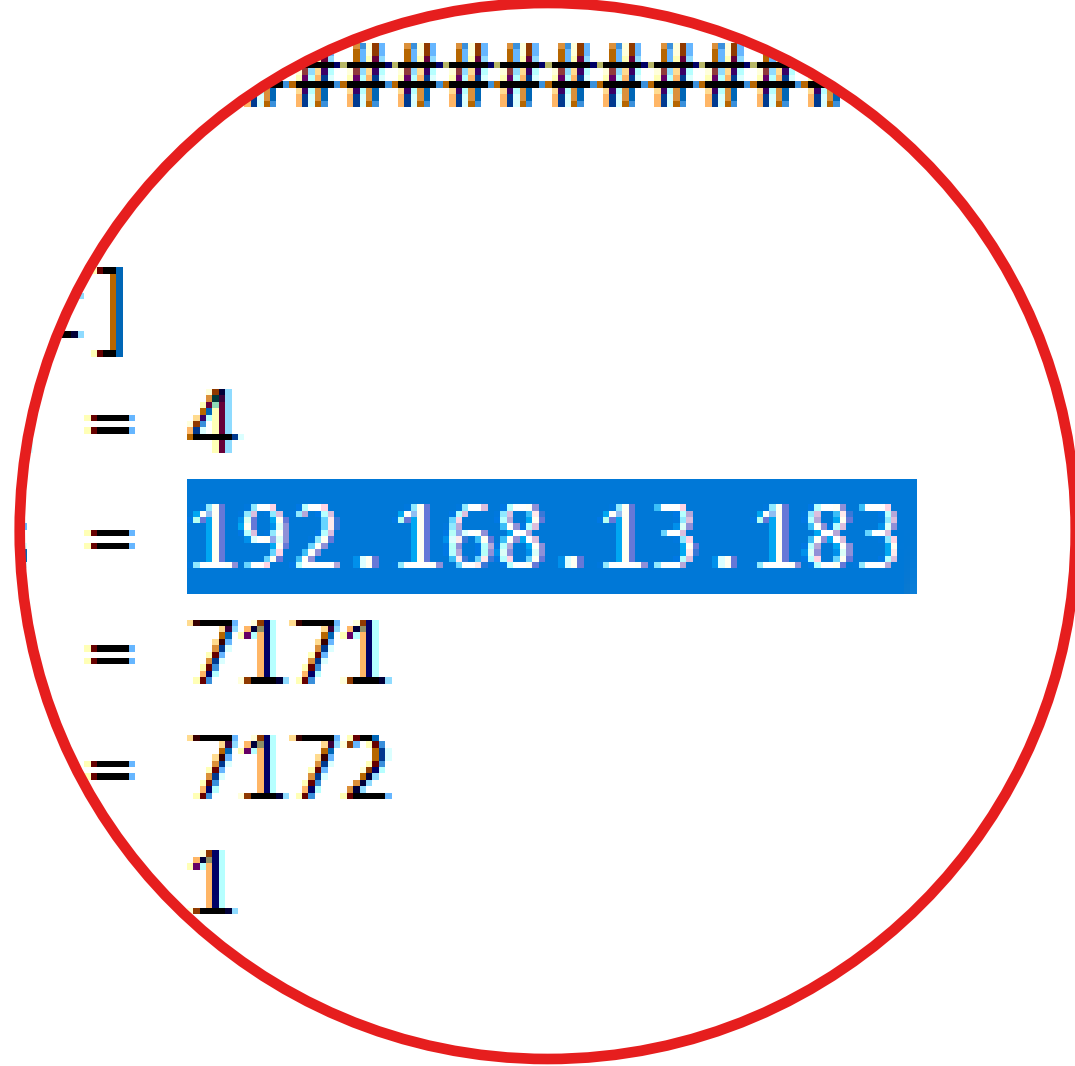
; wlmData.ini example scenario 2 configuration file
; #####

; [LongTerm1.exe]         ; Configuration secti
; version = 4              ; IPv4
; address = 192.168.10.2   ; Instrument server I
; port = 7171              ; Set/Get TCP Port nu
; port2 = 7172             ; CallbackProc/Ex TCP
; offload = 1              ; ConvertUnit / Conve

```

Find the factory set IP address in this file.

11



Replace the factory set IP address by the address of the server-PC.

This information is available by running ipnconfig/all in the command prompt.

12



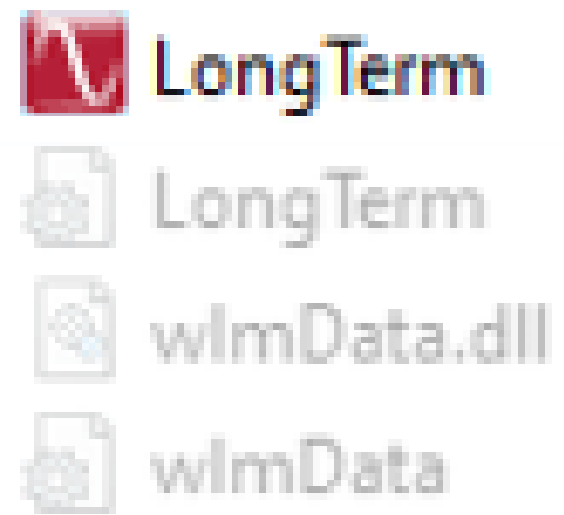
Save ... and close

Save the file **wlmData.ini**  
... and close this file.



13

Open ...



Start the **LongTerm.exe**.

This program allows you to **log the wavelength and additional measurement data**.

It also allows you to **save the data as an ASCII file**.

For a more detailed description of the application see:

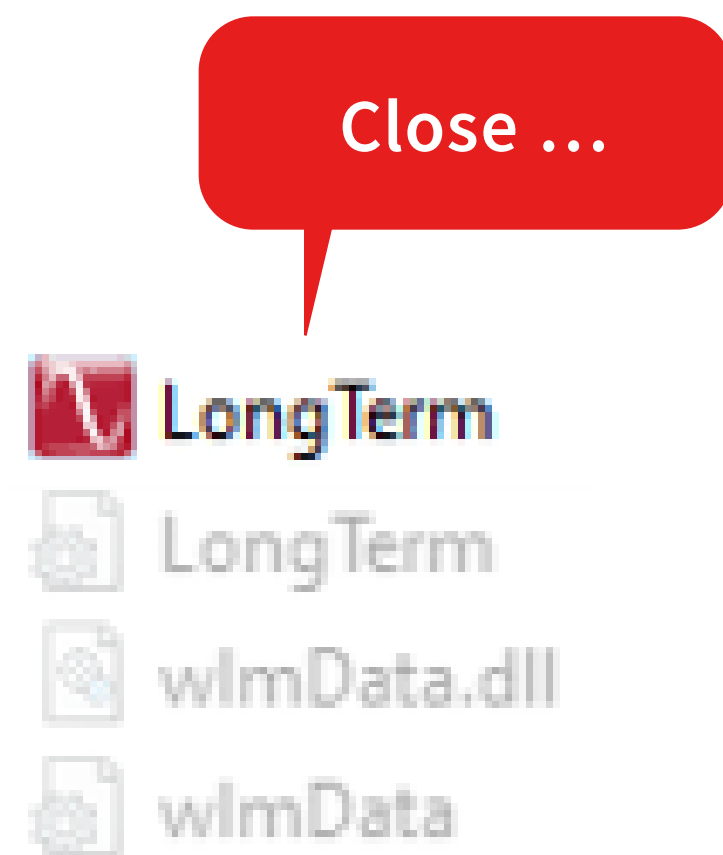
HighFinesse Tutorial

Introduction LongTerm.pdf

The following pages will show you how to ...

... collect the wavelength data using the Callback procedure.

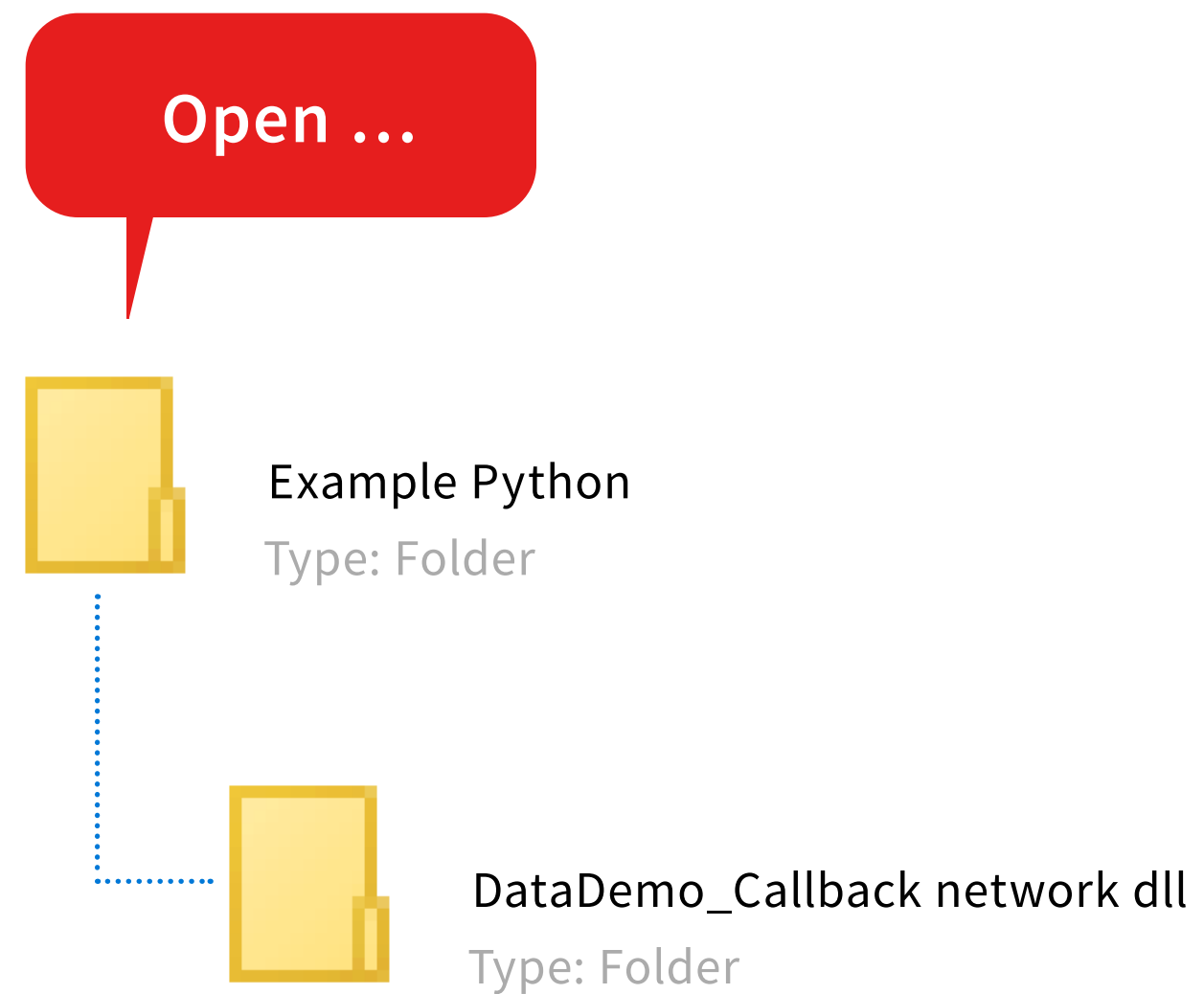
14



Close the **LongTerm** application.

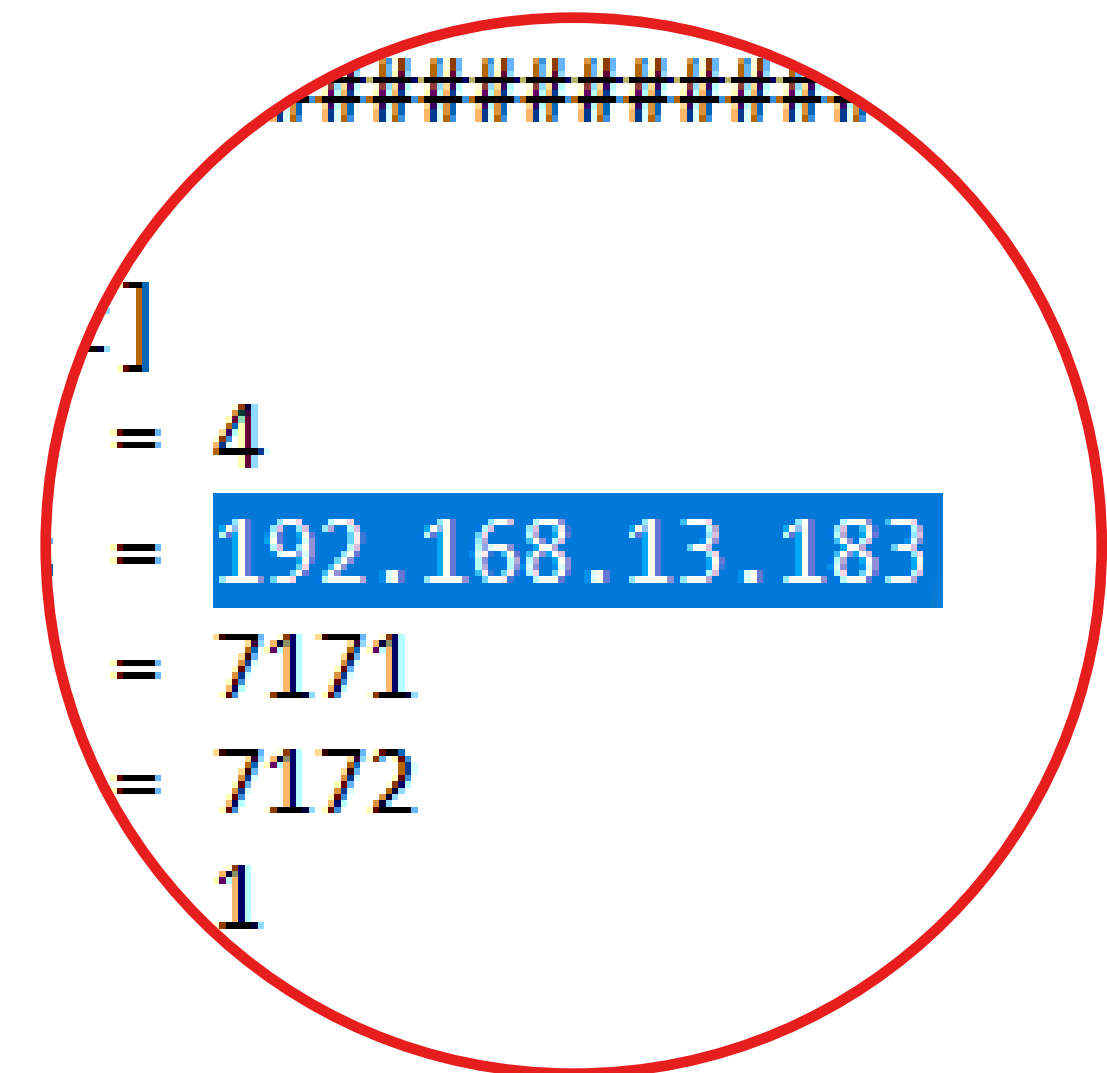


15



Now browse to the folder  
**Example Python/**  
**DataDemo\_Callback network dll**  
and copy it to your client-PC.

16

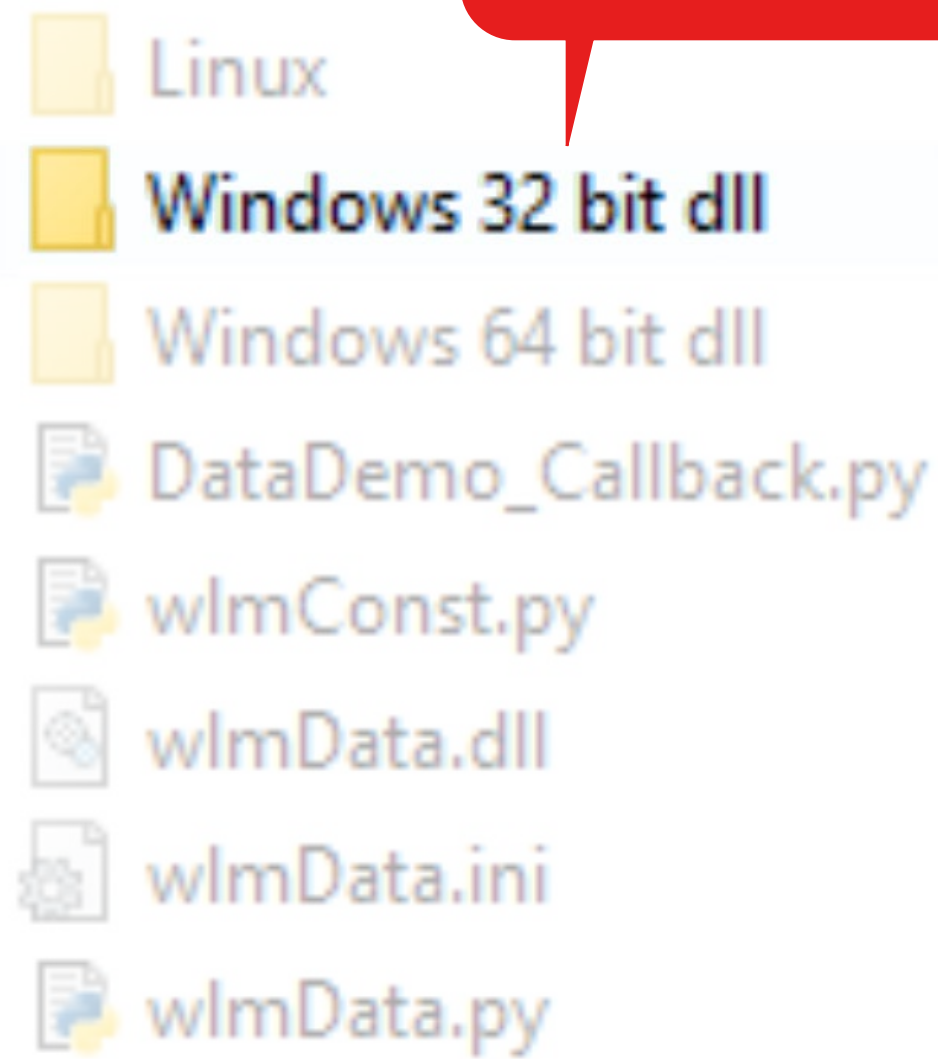


Add the correct **IP address** to  
the **wlmData.ini** just as described  
in step **9**.



17

Open ...



If you using a 32-Python Version replace the **wlmData.dll** by the 32-bit version that is located in the subfolder **Windows 32 bit dll**



18

A screenshot of the Spyder Python IDE. The code editor shows the following Python code:

```

29 import sys
30
31 #####
32 # Set the DLL_PATH variable according to your environment
33 #####
34 DLL_PATH = "wlmData.dll"
35
36
37 # Load DLL from DLL_PATH
38 try:
39     wlmData.LoadDLL(DLL_PATH)
40 except:
41     sys.exit("Error: Couldn't find DLL on path %s. Please check
42
43 # Checks the number of WLM server instance(s)
44 if wlmData.dll.GetWLMCount(0) == 0:
45     print("There is no running wlmServer instance(s).")
46 else:
47     # Read Type, Version, Revision and Build number
48     Version_type = wlmData.dll.GetWLMVersion(0)
49     Version_ver = wlmData.dll.GetWLMVersion(1)
50     Version_rev = wlmData.dll.GetWLMVersion(2)
51     Version_build = wlmData.dll.GetWLMVersion(3)
52     Pattern=wlmData.dll.GetPattern(0)
53     print("WLM Version: [%s.%s.%s.%s]" % (Version_type, Version

```

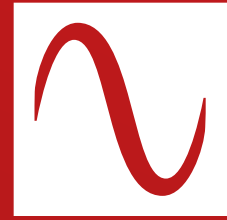
Open **DataDemo\_Callback.py** with a suitable program (for example Spyder).

This script shows you how to efficiently collect the wavelength data using the Callback procedure. This is a good starting point for your own application in Python.

Follow the instruction in the **External Access Guide.pdf** to write your own code.

If you would like to have a deeper understanding on the HighFinesse NetAccess solution please take a look in the ...

... **HighFinesse NetAccess userguide and manual.**



# HighFinesse

The Standard of Accuracy



HighFinesse GmbH  
Wöhrdstraße 4  
72072 Tübingen, Germany



T + 49 (0) 7071 - 53 918 0



F + 49 (0) 7071 - 53 918 99



M [info@highfinesse.com](mailto:info@highfinesse.com)



Additional information  
and distributors:  
[www.highfinesse.com](http://www.highfinesse.com)