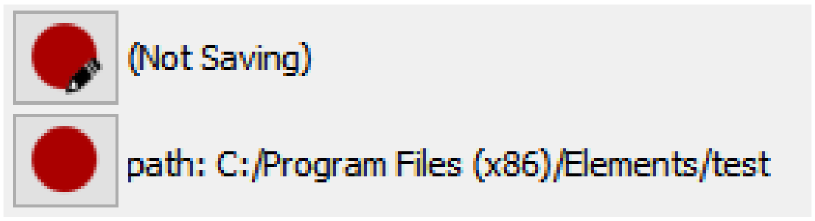
Quick Guide

elements
enabling technologies for Life Science

EDR3 - Record, display and read data



Revision History

Date	Version	Description
14/06/2024	1.3	Added information to the description of the .dat format
09/04/2021	1.2	Updated contacts
05/03/2020	1.1	Revisioned
15/01/2020	1.0	First version of document

Supported data formats

The EDR software can save data in two file formats:

- Raw .dat files format;
- Axon Binary File (.abf) format version 2.0.

All data files are accompanied by an Elements Data Header file (.edh) in order to be loadable by EDA.

Dat format

Binary files in .dat format consist of a continuous series of IEEE 754 single-precision (32-bit) binary floating-point values expressed in little endian (least significant bytes first), with no header. The data is arranged in groups of CurrentChannelsNumber+1 values. Each group consists of CurrentChannelsNumber current samples (ordered by channel number) followed by 1 voltage sample corresponding to a given time instant. Groups are arranged in temporal order.

E.g. for a 4 channels device the values will be:

IChan1Sample1, IChan2Sample1, IChan3Sample1, IChan4Sample1, VChanSample1,
IChan1Sample2, IChan2Sample2, IChan3Sample2, IChan4Sample2, VChanSample2,
and so on.

Every value has no scaling and is expressed in the same unit as the current range selected during the recording, e.g. currents might be expressed in pA or nA. Voltage is always expressed in mV.

Please note that this format does not store intrinsically the current range and the sampling rate information, which are stored in the .edh file, so the user will have to recover those pieces of information from there.

Below is shown a simple python script that loads the data of a 4 channels device and plots it:

```
import struct
import numpy as np
import matplotlib.pyplot as plt

# Parameters
CurrentChannelsNumber = 4
data_file = 'file_000.dat'
samplingRate = 100000  # in Hz, assuming a sampling rate of 100kHz

# Function to read binary data
def read_binary_file(file_name, channels):
    data = []
    with open(file_name, 'rb') as f:
        while True:
            try:
                # Read a group of channels + 1 (voltage) samples
                   group = struct.unpack('<' + 'f' * (channels + 1), f.read(4 *
(channels + 1)))
                data.append(group)
            except struct.error:
                break
    return np.array(data)

# Read the data from the binary file
data = read_binary_file(data_file, CurrentChannelsNumber)

# Separate the current and voltage data
currents = data[:, :CurrentChannelsNumber]
voltage = data[:, CurrentChannelsNumber]

# Generate the time axis based on the sampling rate
time = np.arange(0, len(voltage)) / samplingRate

# Plotting
fig, (ax1, ax2) = plt.subplots(2, 1, figsize=(15, 10))

# Plot current traces
for i in range(CurrentChannelsNumber):
    ax1.plot(time, currents[:, i], label=f'Current Channel {i+1}')
ax1.set_title('Current Traces')
ax1.set_xlabel('Time (s)')
ax1.set_ylabel('Current (nA)')
ax1.legend()

# Plot voltage trace
ax2.plot(time, voltage, label='Voltage', color='red')
ax2.set_title('Voltage Trace')
ax2.set_xlabel('Time (s)')
ax2.set_ylabel('Voltage (mV)')
ax2.legend()

# Show the plot
plt.tight_layout()
plt.show()
```

Abf format

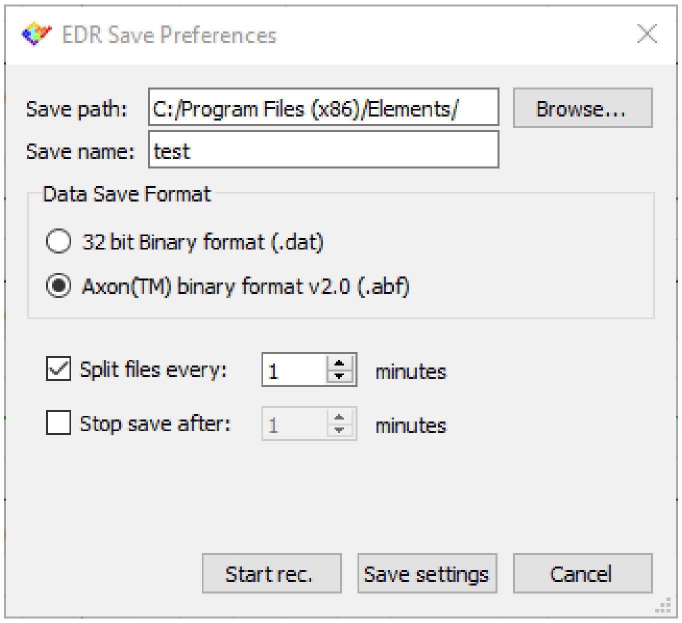
It's the Axon file format used by pClamp software.

Data are saved in "gap-free" acquisition mode.

Save preferences

After connecting the device, EDR software starts to acquire and display data in real time in the main window.

To store data, users can either click the Setup save settings and start recording button to enter into the Save Preferences menu and then click “Start rec.”, or can quickly start a recording by clicking the Quick Save button .



Save path: C:/Program Files (x86)/Elements/
Save name: test

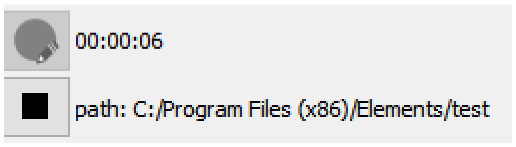
( ) 32 bit Binary format (.dat)
[✓ Axon(TM) binary format v2.0 (.abf)]

[✓] Split files every: 1 minutes
[ ] Stop save after: 1 minutes

Ctrl + ‘G’ keyboard shortcut can also be used to quickly start and stop data recording.

Multiple records will be saved with the set name followed by the progressive number (e.g. C:/Program Files (x86)/Elements/test_01, ...test_02, ...test_03.... and so on) .

When recording begins, a timer in the GUI starts and indicates the total recording time.



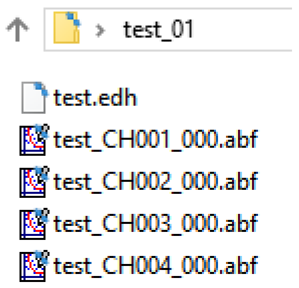
00:00:06
path: C:/Program Files (x86)/Elements/test

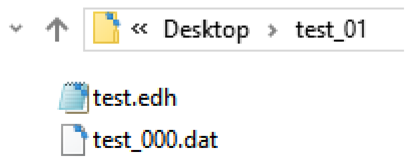
To stop recording, click on the Stop recording button .

The EDR software creates a new folder in the location chosen by the user with the same name of the recording files. In this new folder the EDR software creates some files:

- An header .edh file (which is a text file containing some acquisition setup information, such as Range, Bandwidth, etc.).
- The saved data files.

For multichannel devices, if .abf format is selected, a different .abf file will be created for every current channel, while a single file .dat file will be created if .dat format is selected.

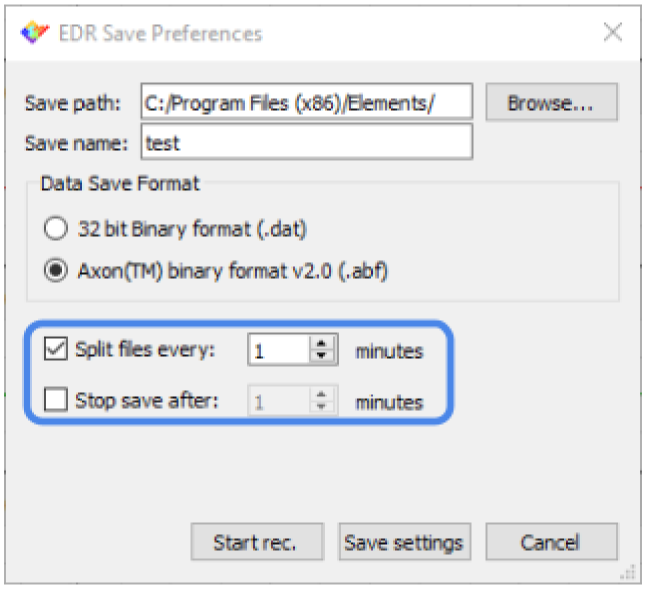

test_01
test.edh
test_CH001_000.abf
test_CH002_000.abf
test_CH003_000.abf
test_CH004_000.abf


« Desktop › test_01
test.edh
test_000.dat

Be careful to have enough disk space to save data and avoid recording data directly into network drive or Cloud storage service. Notice that a single channel device at the maximum sampling rate of 200kHz stores about 50MB/min using .abf format, and 100MB/min using .dat format.

Split of data files and automatic save stop

To limit the maximum file dimensions, the EDR software can splits the data files every X minutes of recording. This feature can be selected from the "Save Preferences" dialog box and can be useful during long recording, in order to obtain smaller files, easier to load in offline analysis tools.


EDR Save Preferences

Save path: C:/Program Files (x86)/Elements/

Save name: test

Data Save Format

- 32 bit Binary format (.dat)
- [✓ Axon(TM) binary format v2.0 (.abf)]

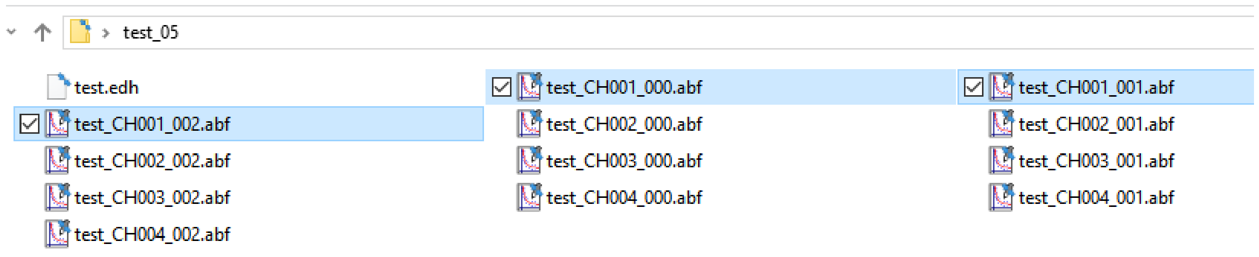
[✓] Split files every: 1 minutes

[ ] Stop save after: 1 minutes

Also .abf files can be split in small chunks during data recording.

The Save Preference dialog box also offers an option to automatically stop the recording after a fixed amount of time.

The following image shows a folder of an example recording performed using an e4 (4 channels device), which lasted slightly more than 2 minutes, with files split every 1 minute:



test_05

test.edh
test_CH001_002.abf
test_CH002_002.abf
test_CH003_002.abf
test_CH004_002.abf
test_CH001_000.abf
test_CH002_000.abf
test_CH003_000.abf
test_CH004_000.abf
test_CH001_001.abf
test_CH002_001.abf
test_CH003_001.abf
test_CH004_001.abf

Read Data

The raw .dat files can be handily read by EDA (Elements Data Analyzer), or by Matlab with the function elementsRead.m file, or by Python using the “elementsRead.py”. All these programs can be found in the EDR installation folder (default path: C:\Program Files (x86)\Elements).

Abf files saved byu EDR can be read by EDA as well and are compatible with all softwares that can load abf files.