

TCP Streaming Interface

Document Revision: 1.6



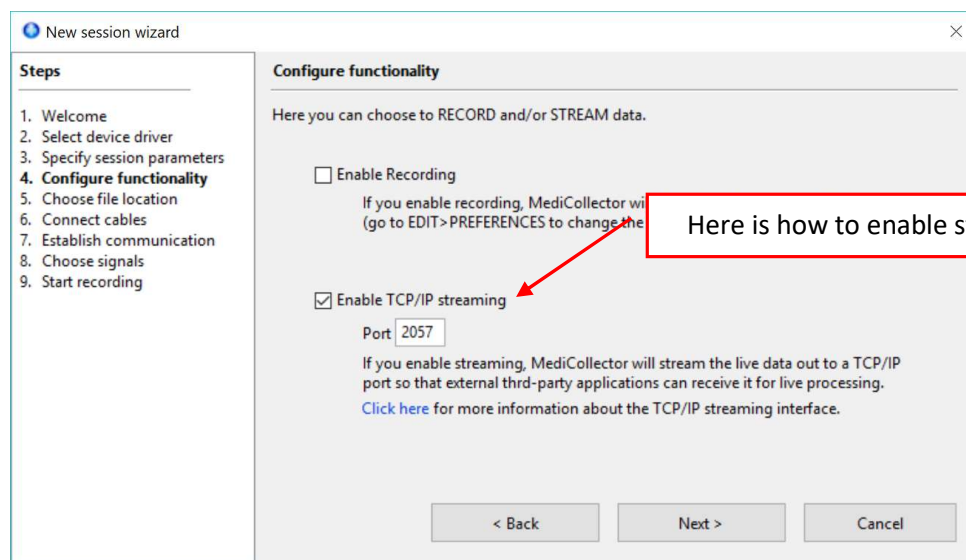
Purpose

As of the release of **v1.0.46** of MediCollector BEDSIDE, users can now stream live data out via TCP/IP.

This TCP/IP interface is intended for developers who want MediCollector to send live data to an external application or system. Using this interface, you can send live signals from one of our supported medical devices directly into your own application.

Please note the following:

- This interface is intended for advanced programmers. Do not attempt to use this interface if you are not skilled in a programming language which can receive TCP/IP data. MediCollector does not provide technical support for your programming!
- To enable streaming in MediCollector, you just need to turn it ON when you launch a new recording session. This is done when you launch a new session as seen in screenshot below. Its easy! Just turn it ON and select the port.
- Live data may be delayed by a short time lag (e.g. about 1 second on Philips Intellivue devices). This is unavoidable and caused by the medical device itself (not MediCollector). By the time MediCollector receives data from a medical device, it is already ~1 second delayed.
- MediCollector will only stream to a port on the localhost. And will only allow 1 client at a time.
- Data that is sent using this interface is NOT encrypted. However, no patient identifiers are sent either.
- Data is sent in “packets” using the protocol defined on the following page. Each packet will contain a variable number of datapoints for a specific signal. It is up to you to unpack these signals and use them as you wish.
- Example programs written in Matlab and LabVIEW are provided with this interface. This example programs will receive the stream of data and unpack and plot the signals. These examples are a good place to start!



Protocol

MediCollector will send data packets as soon as they are received from the medical device. Each packet contains a varying number of data datapoints for a signal. Each packet is formatted as seen below. There are 4 fields in each packet. The first 3 fields have a fixed width as specified below. The last field contains the actual datapoints and has a varying length depending on the number of datapoints.

[SignalName][SignalFrequency][Tzero][Datapoints][TerminationChar]

Field	Field Length	Description
[SignalName]	30 chars	The name of the signal contained in this packet, padded with blank spaces to fill the 30 chars width.
[SignalFrequency]	10 chars	The frequency of the signal in Hz, , padded with blank spaces to fill the 10 chars width. This value always uses "." as the decimal point.
[Tzero]	23 chars	The timestamp of the first datapoint in this packet in the following fixed format "YYYY.MM.DD HH:MM:SS.000" This field is ALWAYS 23 chars in length. It always uses "." as the decimal point. And it always uses 24 time (aka military time) and has 3 values after the decimal point.
[Datapoints]	Varies	An array of 8-byte doubles
[TerminationChar]	5 chars	Always equals ' '

To process packets, you should build a program which waits for a termination character (| | | |) and then processes each packet according to the specs above. To determine how many datapoints are in each packet (because it will vary), you need to divide the length of the Datapoints field by 8 (because there are 8 bytes in each datapoint).

Below is an example TCP packet:

"SlowRandom 0.976562 2017.09.02 14:39:17.863pTf'r¶#@q'X²~i"@| | | |"

The corresponding field values are:

SignalName = SlowRandom
SignalFrequency = 0.976562 Hz
Timestamp = 2017.09.02 14:39:17.863
Datapoints = Two values: 9.85634 and 9.46421
In hex, these 16 bytes are: FE54 A327 72B6 2340 71B9 58B2 ACED 2240

Examples

Example programs written in Matlab and LabVIEW are installed with the MediCollector BEDSIDE product. If you run the example code, it will receive the stream of data and unpack and plot the signals. The examples are a good place to start! The example files are usually found in the following folder, depending on your installation location:

C:\Program Files\MediCollector\Streaming