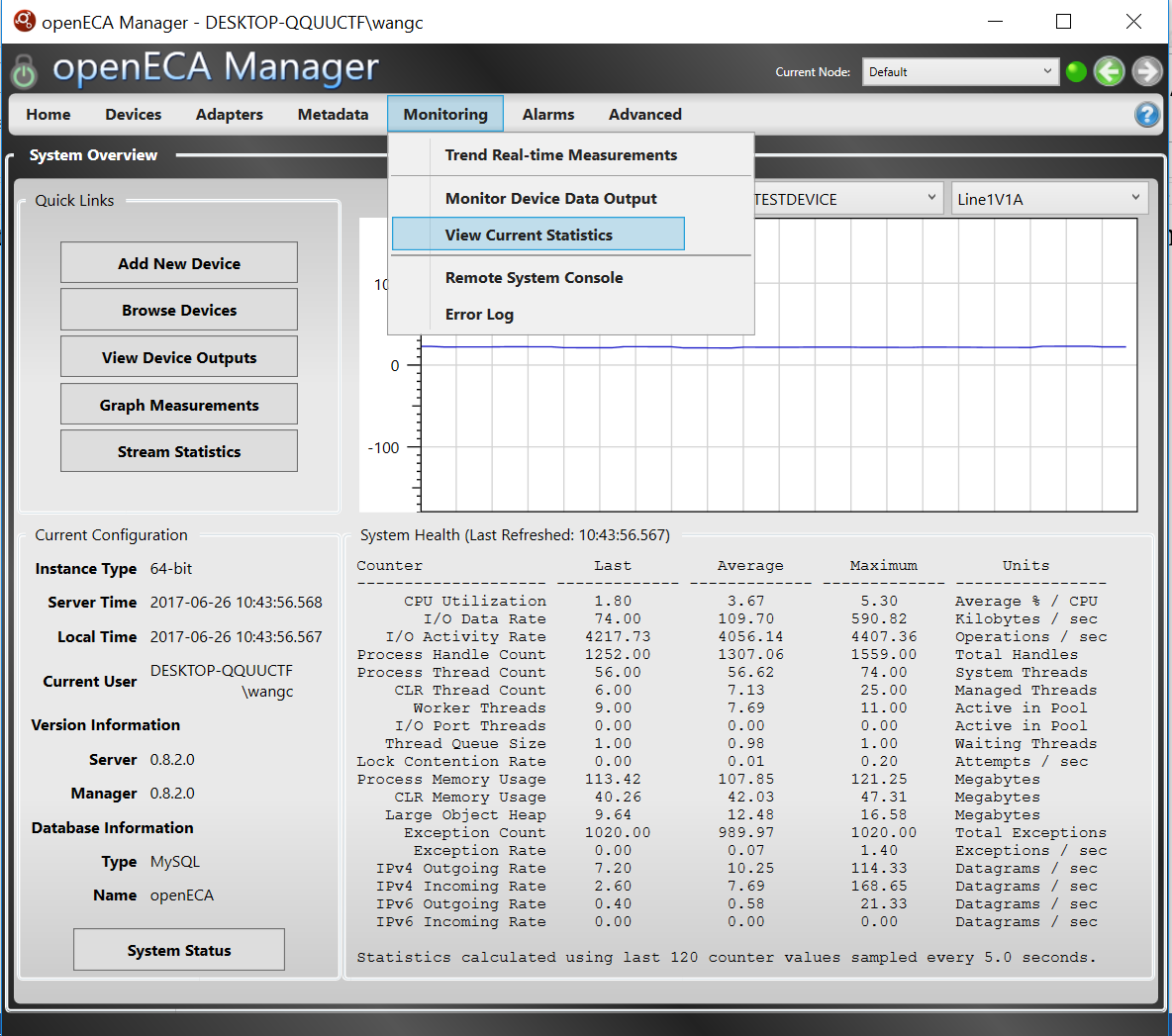
**openECA Streaming Data Collection Through openHistorian**

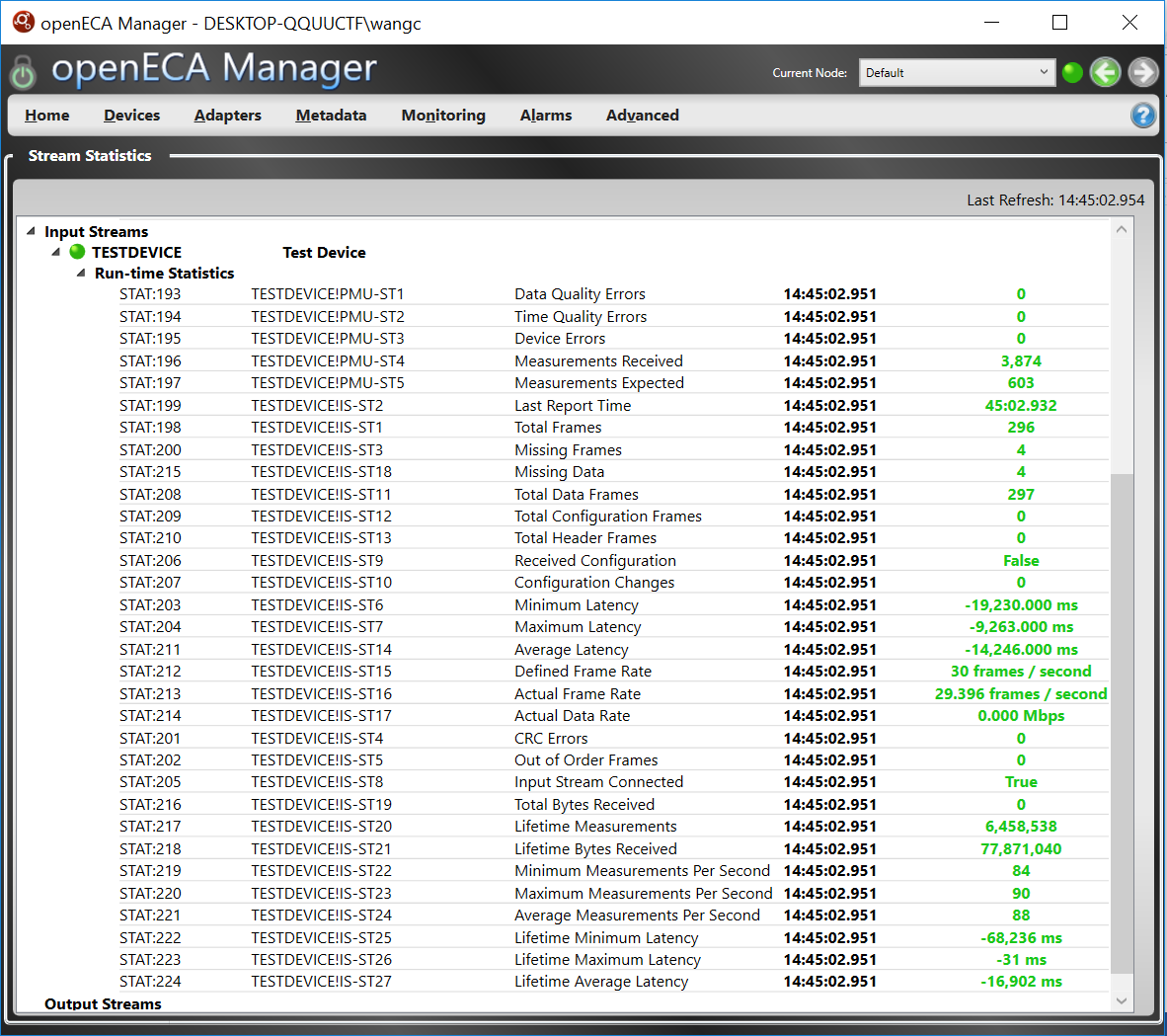
**Chen Wang**

The aim of this documentation is to show the steps that one could follow to acquire the openECA streaming data during a user-specified past time period through an off-line platform, i.e. openHistorian. Such functionality can provide off-line applications based on openECA platform with usable data set formatted with defined structure. The process is composed of, firstly, configuring openHistorian with proper data archive (assuming that openECA has been properly configured.); secondly, building up internal connections between openECA and openHistorian; thirdly, utilizing openHistorian API to acquire the data with proper settings. The details are shown in the following parts.

**STEP 1: Configuring openHistorian**

1. Find ‘Monitoring’ ̶> ‘View Current Statistics’, make sure there are data streaming in openECA.





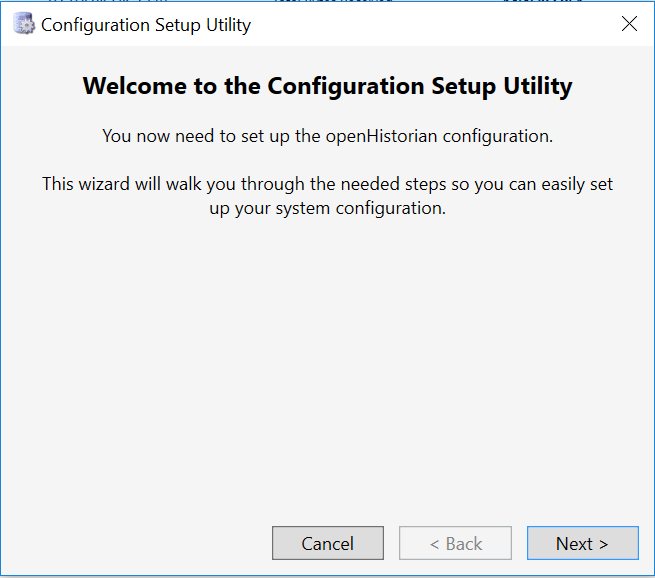
1. Download and install openHistorian.

The place to download the install file can be found in the following page:

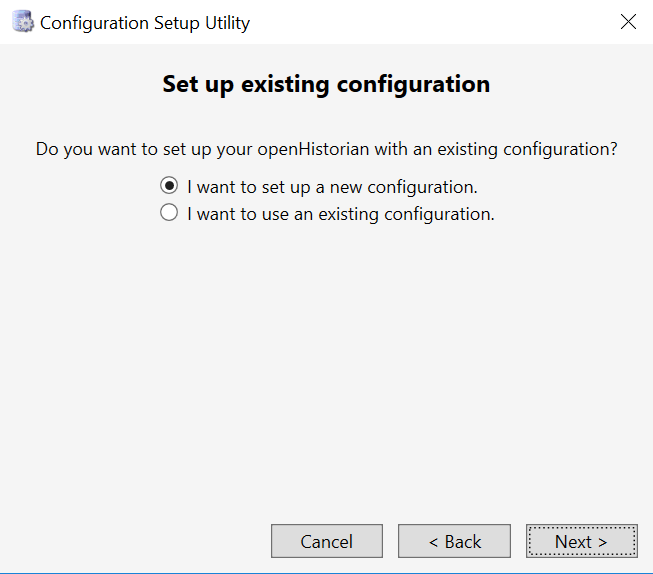
https://github.com/GridProtectionAlliance/openHistorian

1. Configure the openHistorian

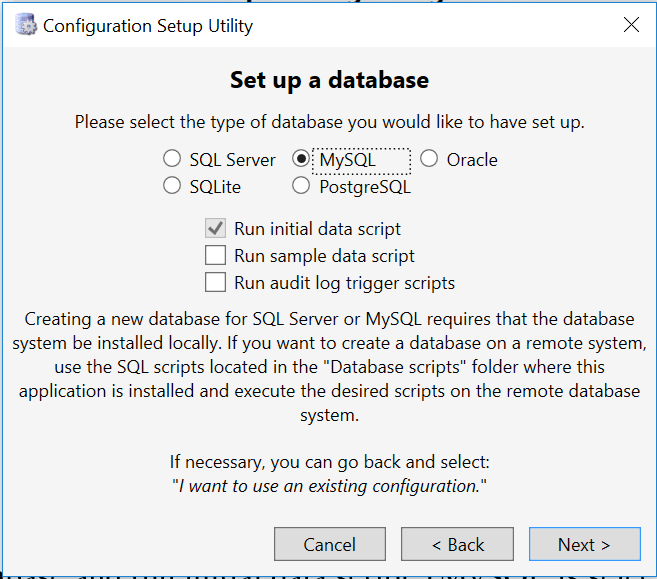
Launch the ConfigurationSetupUtility



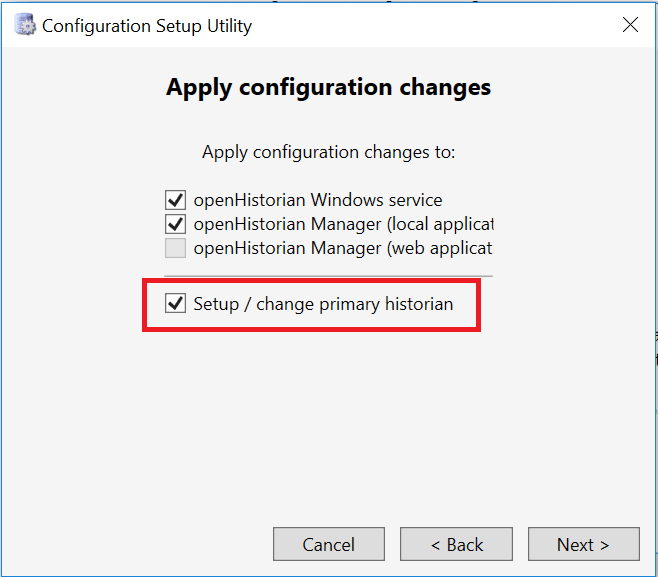
Set up a new configuration

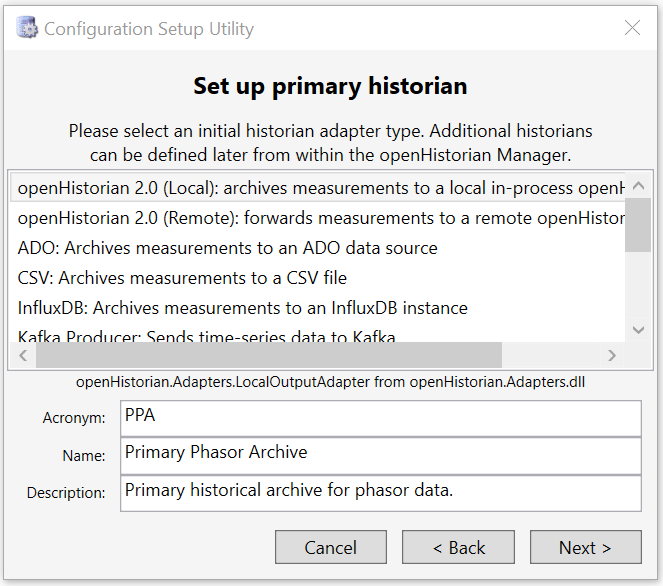


Select using Database and run initial data script. (MySQL is selected here just to be consistent with the openECA configuration. Other options are acceptable too.)



When apply the configuration settings, do check the ‘Setup/change primary historian’.



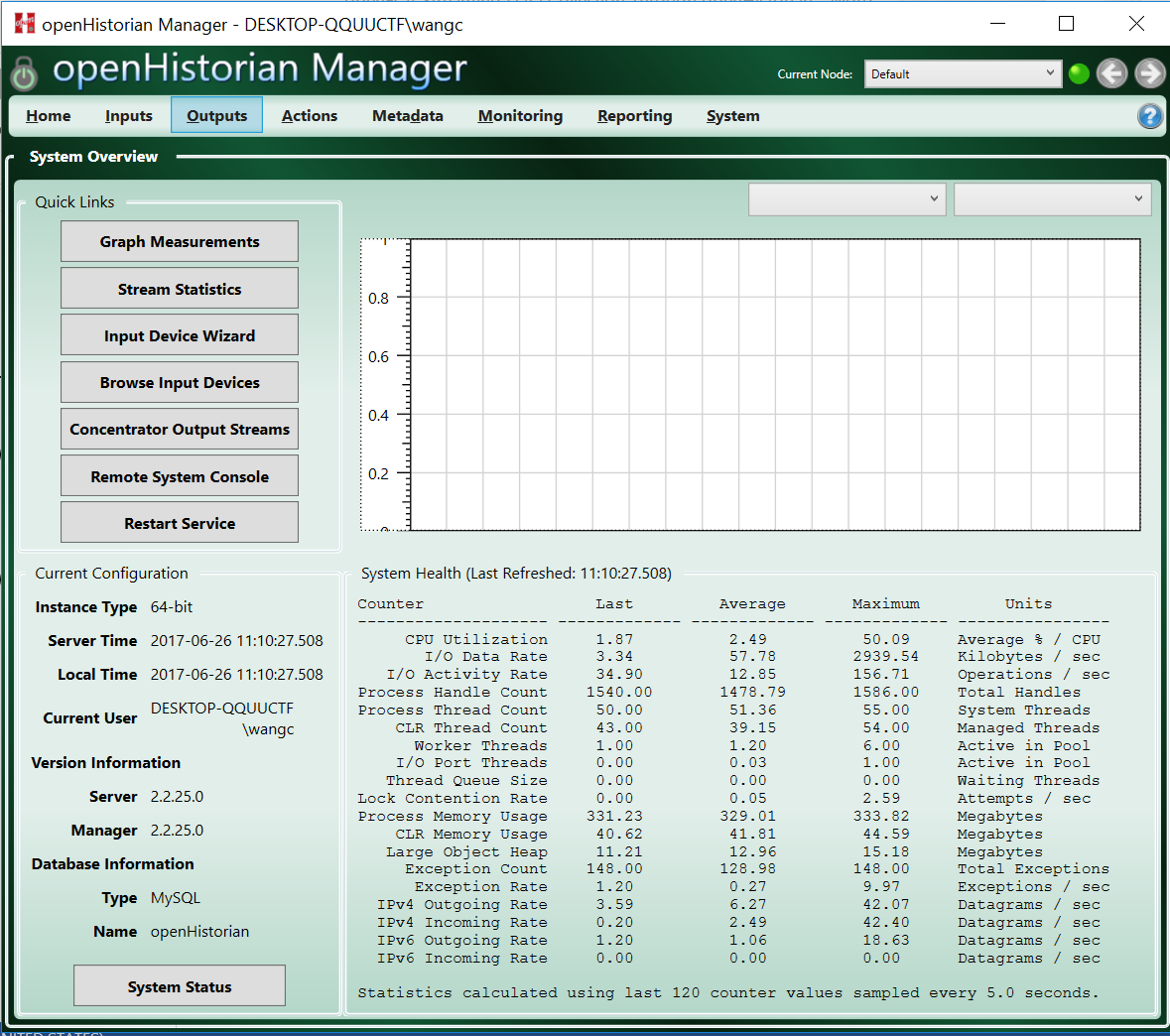


**STEP 2: Building up internal connections between openECA and openHistorian**

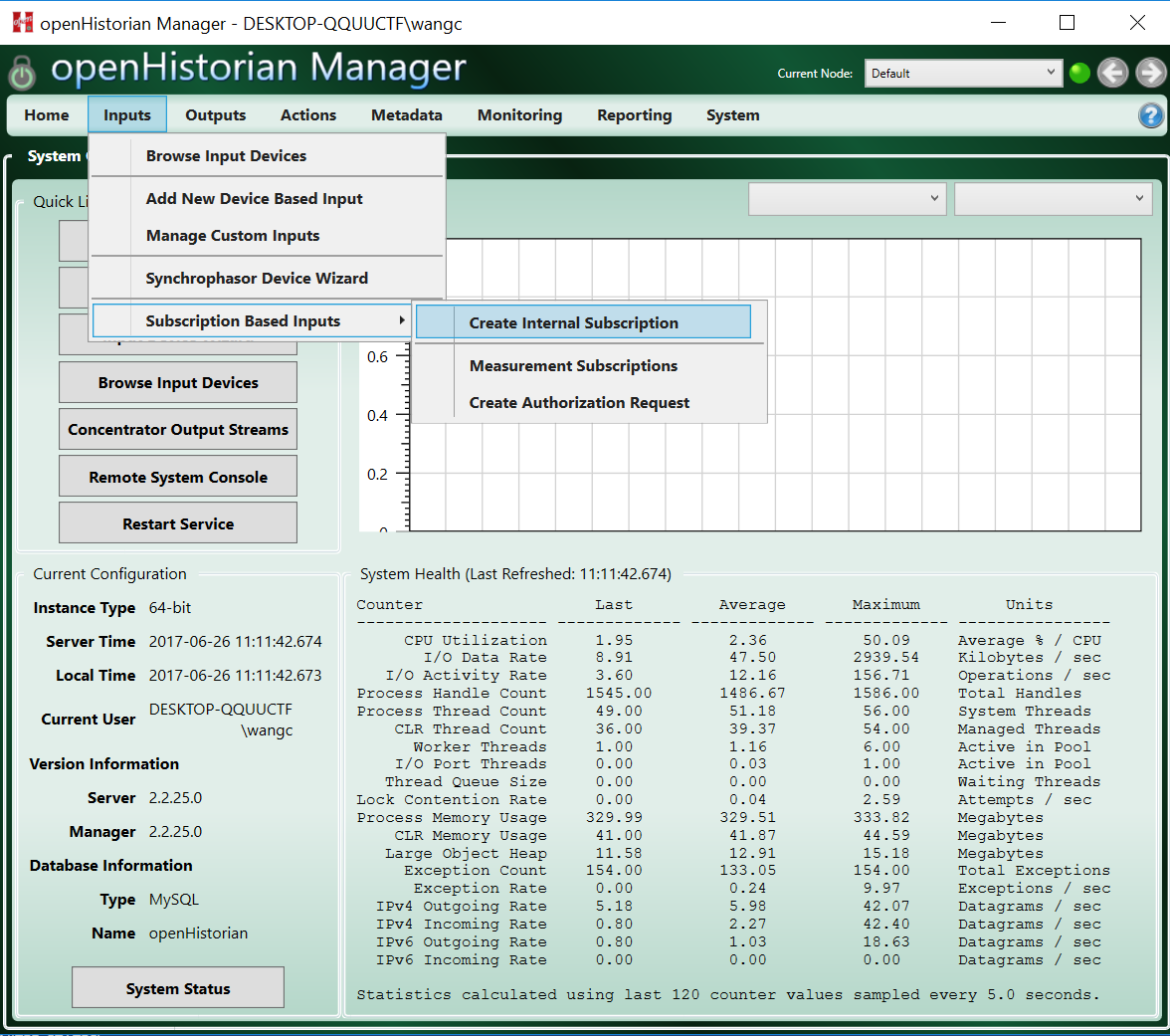
**[One of the good references for this step could be found here:**

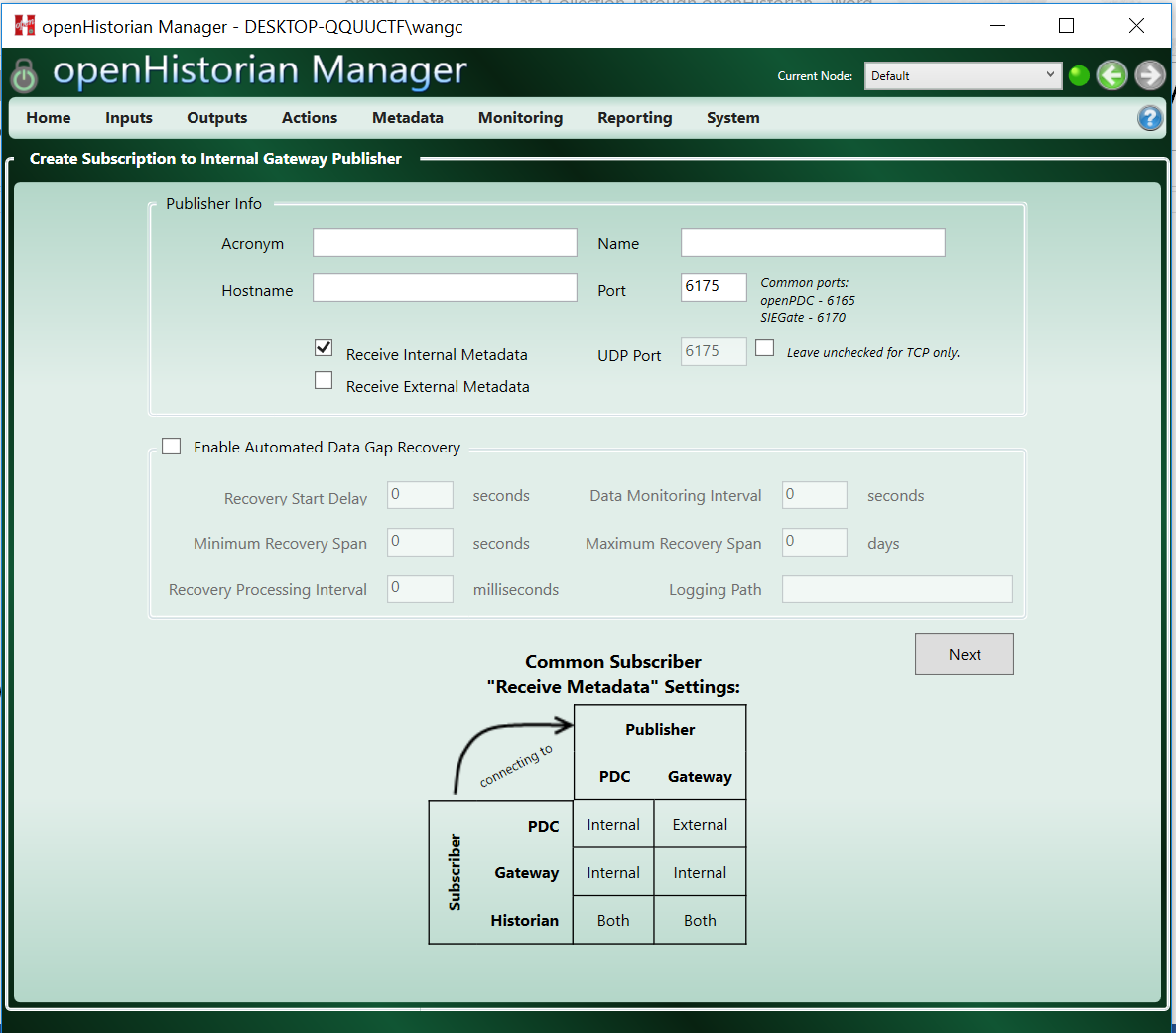
[**https://github.com/GridProtectionAlliance/SIEGate/blob/master/Source/Documentation/wiki/Creating\_Internal\_Gateway\_Connections.md**](https://github.com/GridProtectionAlliance/SIEGate/blob/master/Source/Documentation/wiki/Creating_Internal_Gateway_Connections.md) **]**

1. Launch openHistorianManager (May require running as administrator)



1. Find ‘Inputs’ ̶> ‘Subscription Based Inputs’ ̶> ‘Create Internal Subscription’

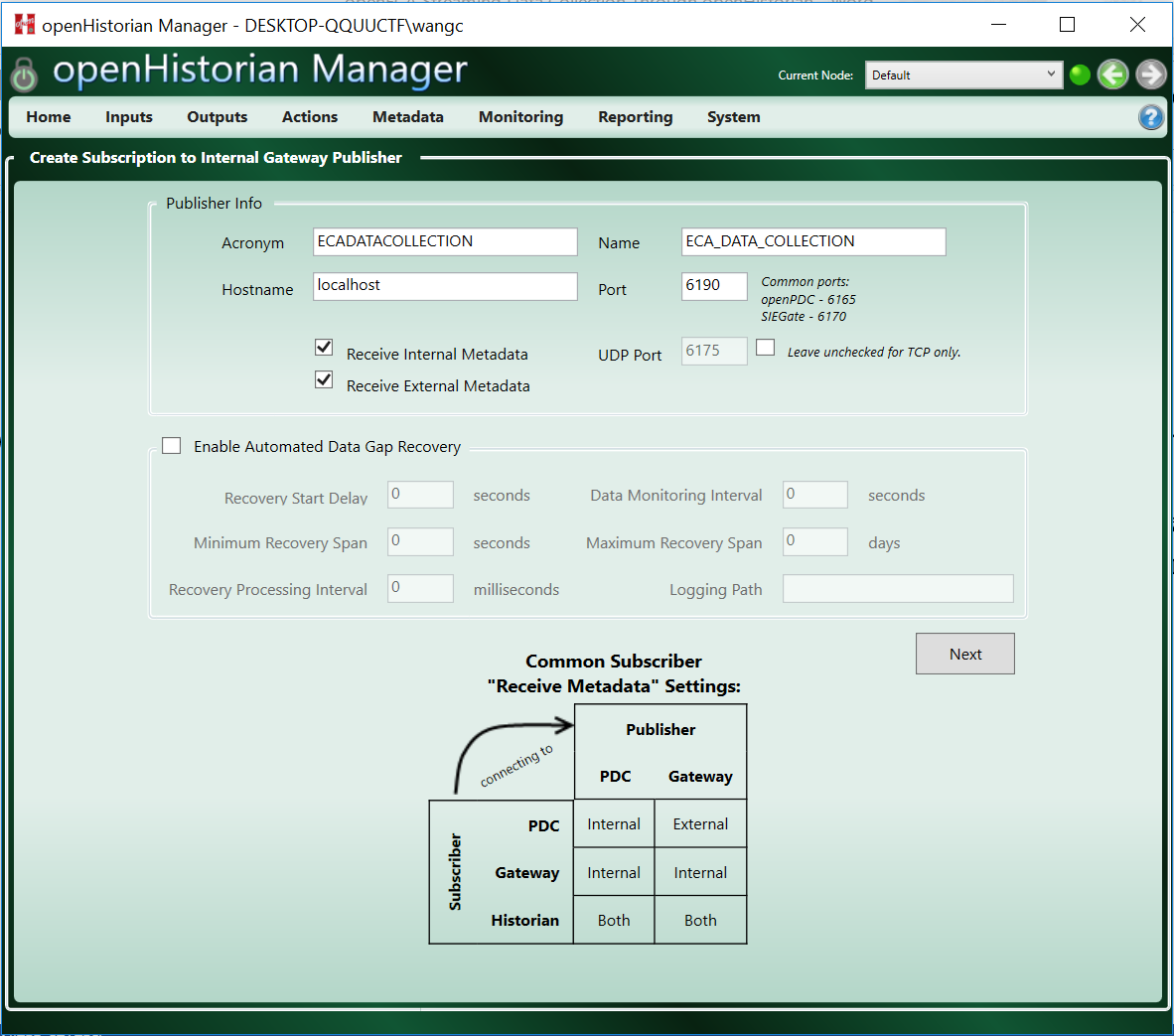




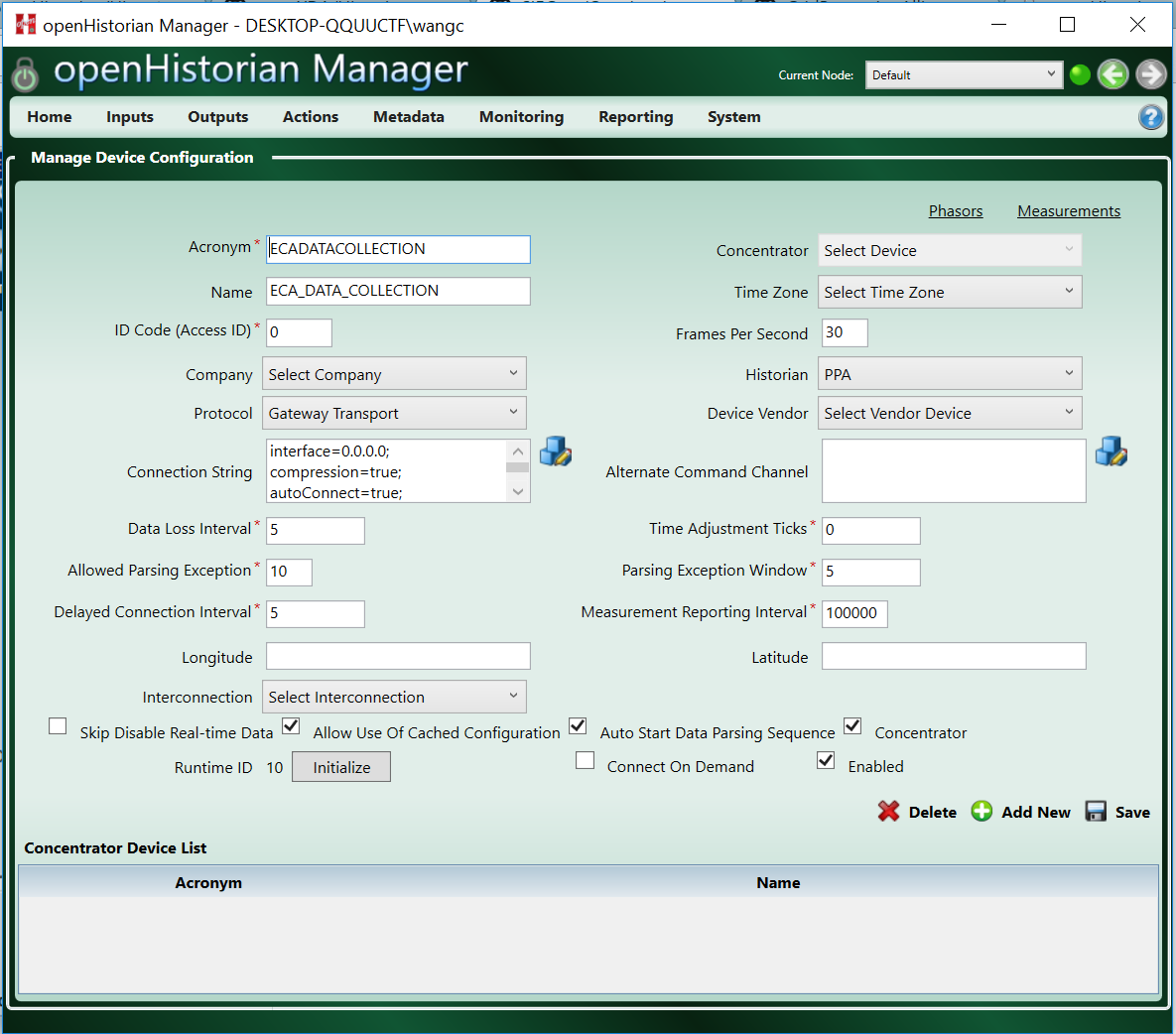
1. There are five things to be set: Acronym, Name, Hostname, Port, and which data to receive.

[Explanations of those names can be found in the references aforementioned.]

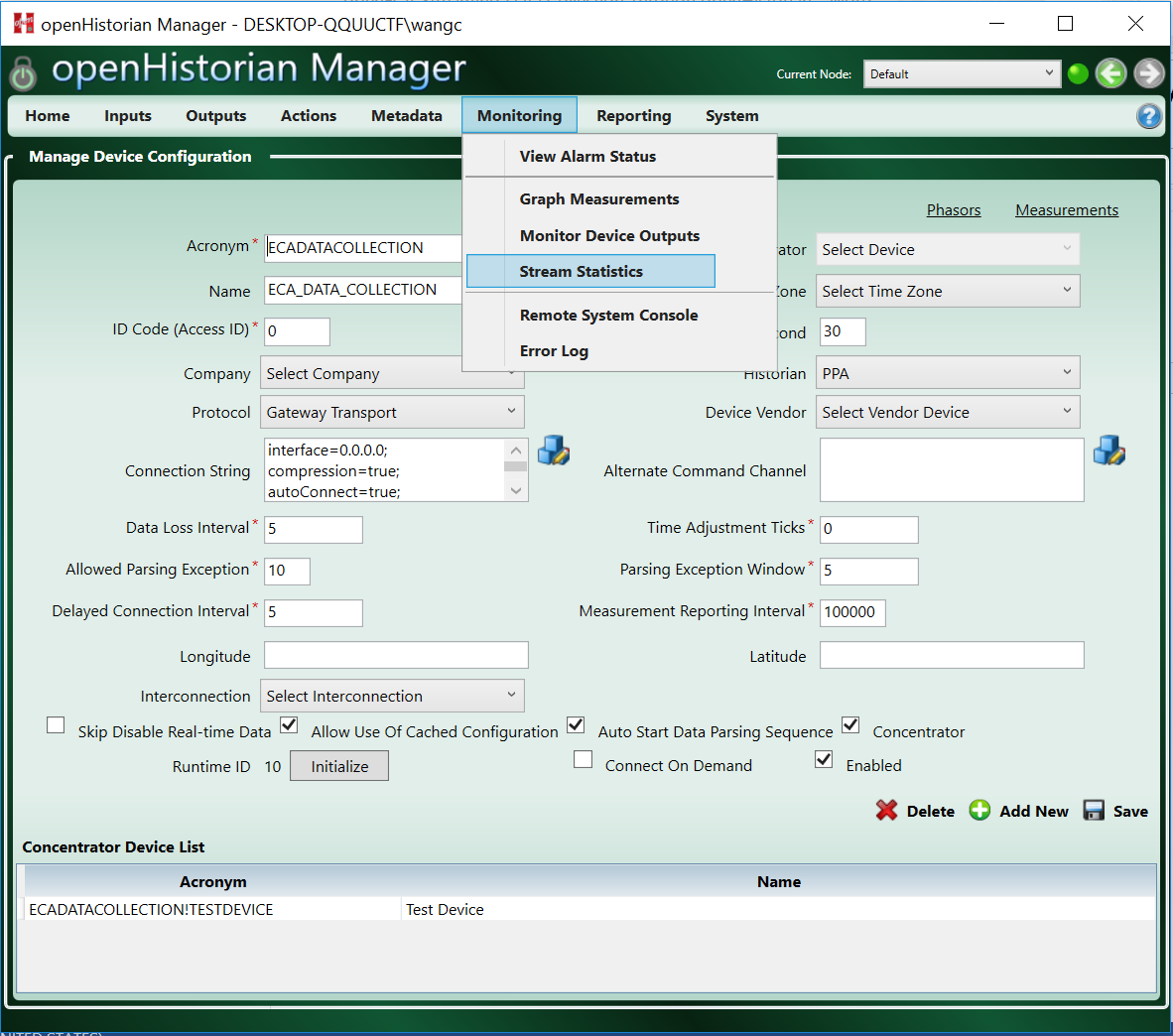
Acronym and Name can be defined by user. Hostname should be set as ‘localhost’ if openECA and openHistorian are installed on the same computer. Port should be 6190 which indicate openECA’s default port. ‘Receive Internal Metadata’ and ‘Receive External Metadata’ should both be checked. The completed setting example is as shown in the following figure.

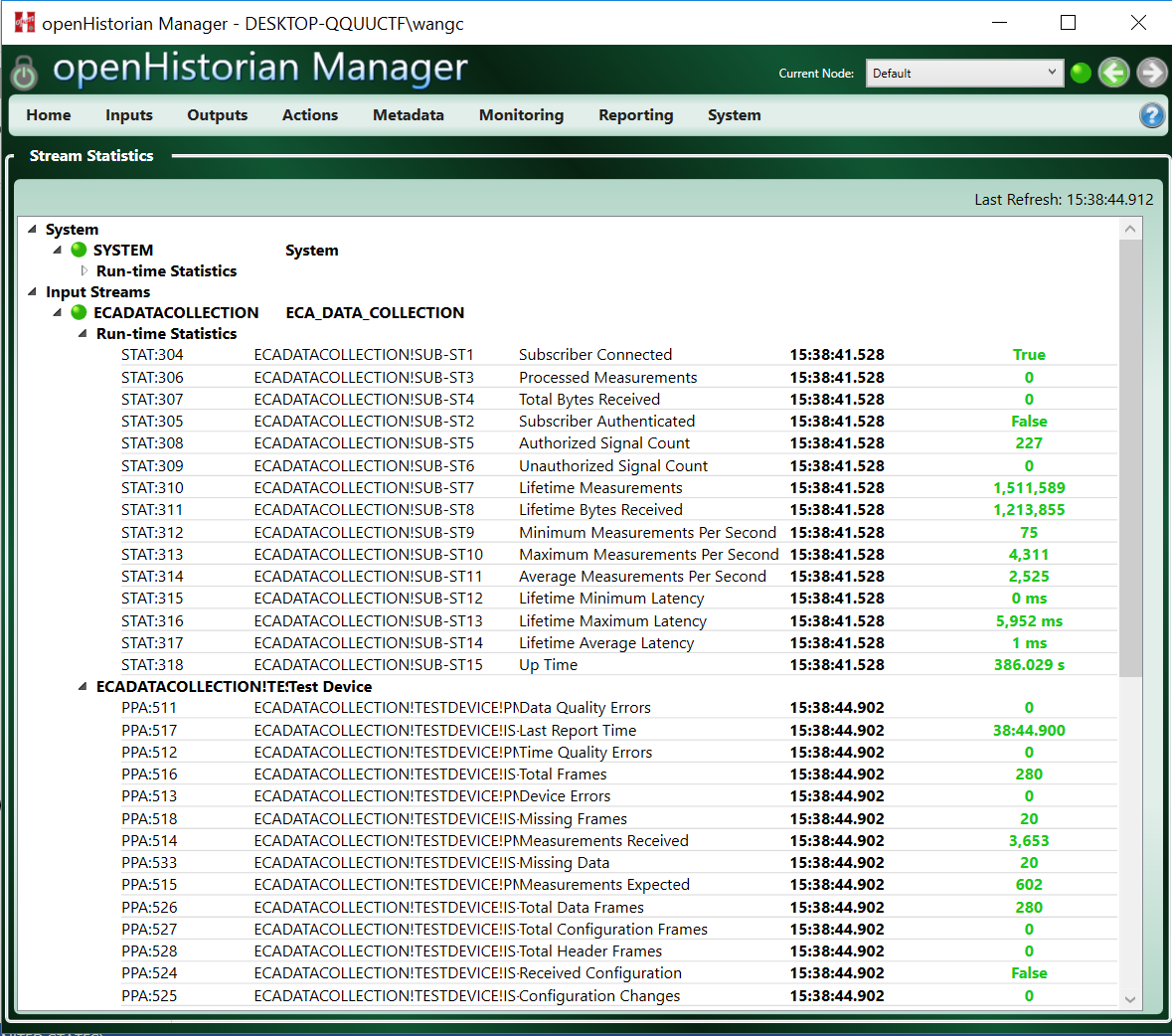


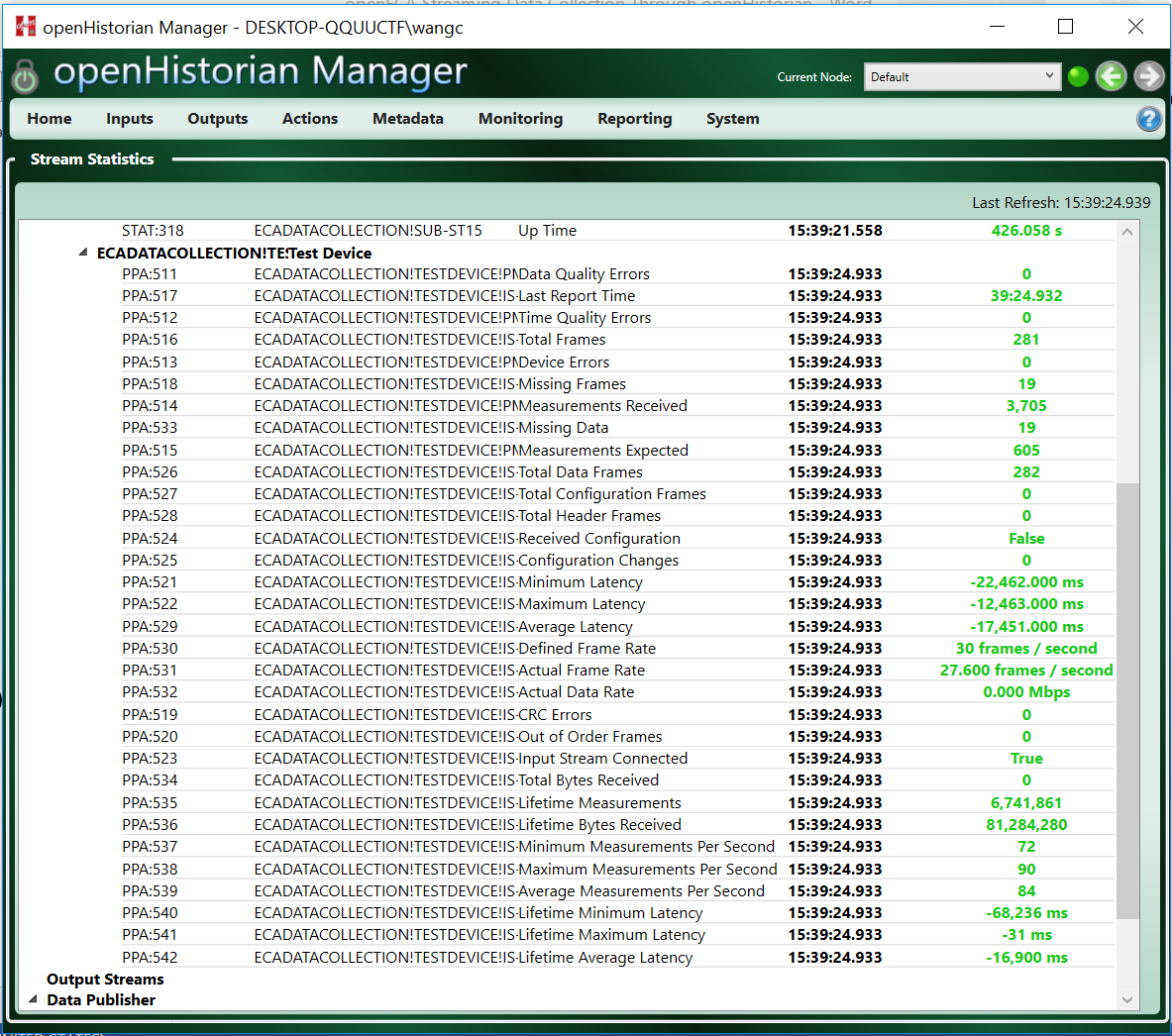
1. After clicking ‘Next’, the device configuration page will show up. Click ‘Save’ if no other settings needed. Then click ‘Initialize’ to complete the building of the connection.



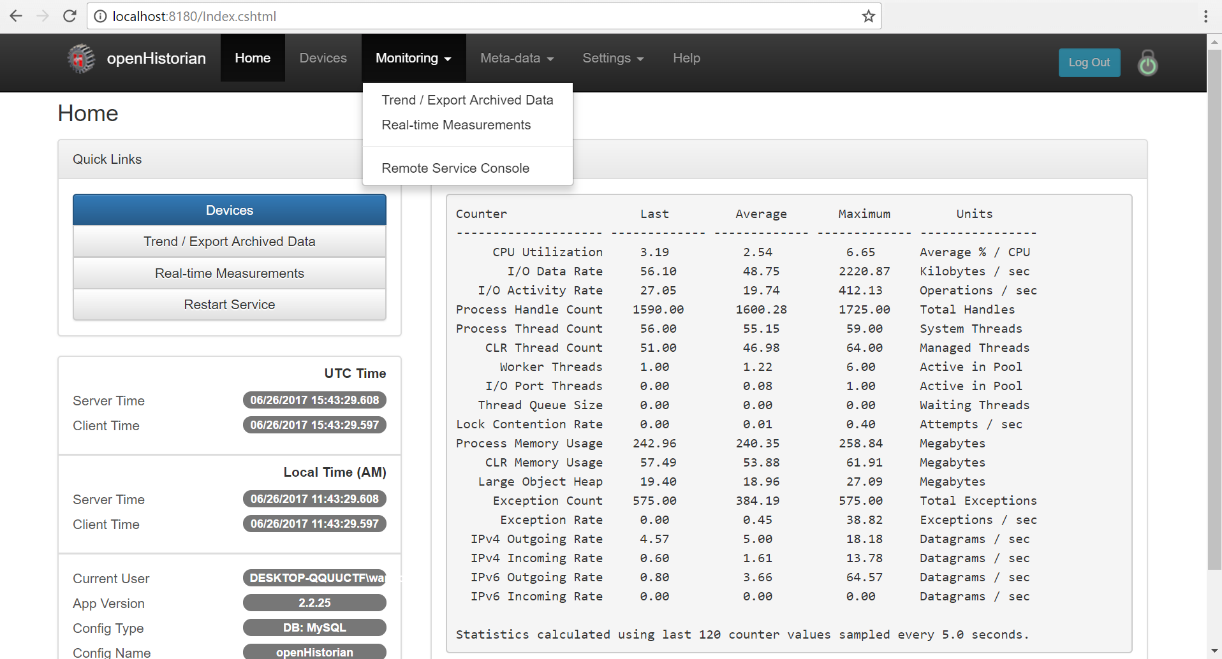
1. Connection Test 1: Check ‘Monitoring’ ̶> ‘Stream Statistics’

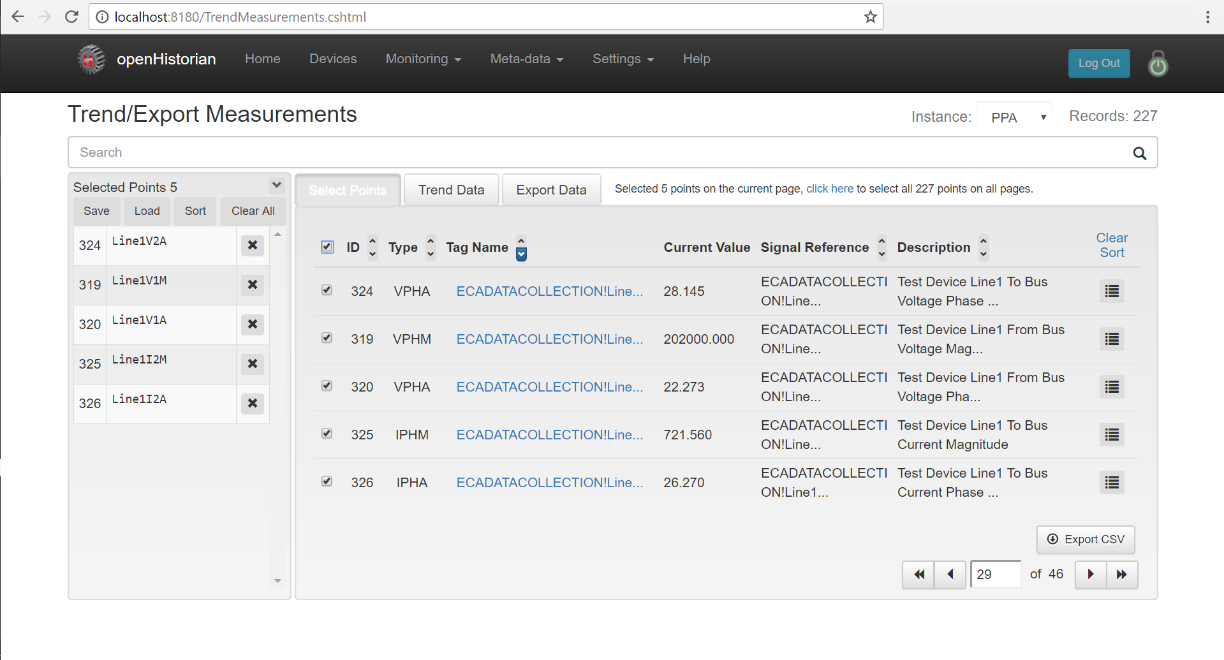


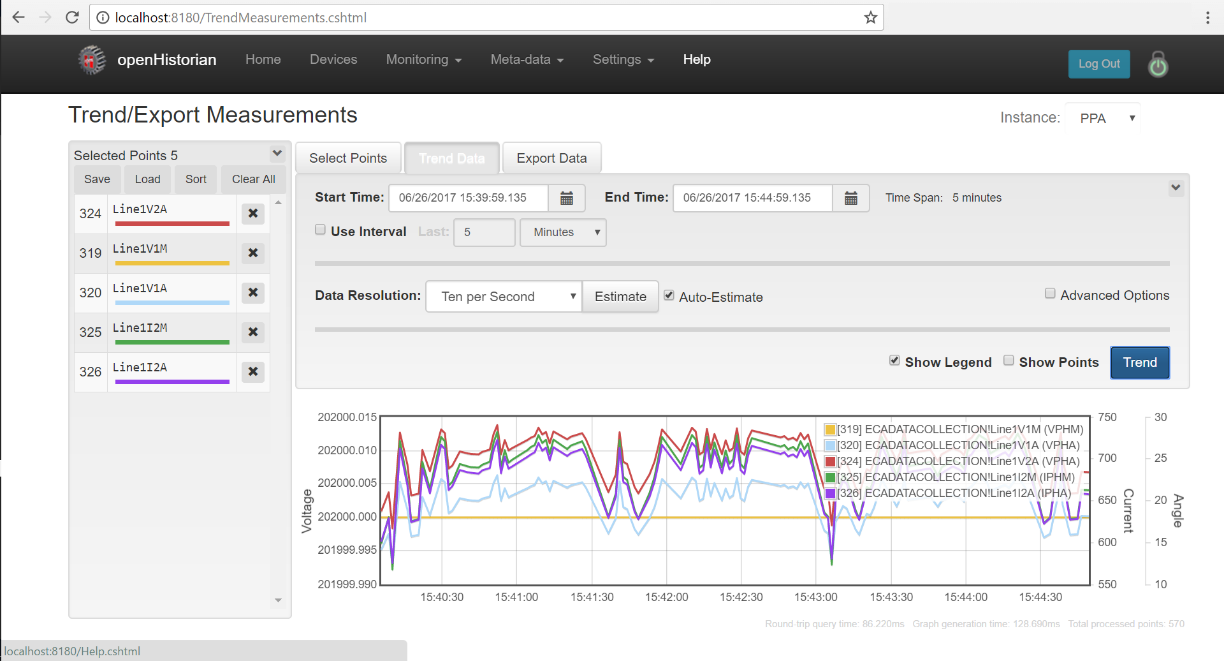




1. Connection Test 2: Try to Trend data on the Web tool.







**STEP 3: Utilizing openHistorian API to acquire off-line data set**

1. Conduct the data acquisition using the proposed C# based application. The default query settings should be as following:

Host Address: 127.0.0.1

Data Port: 38402

Metadata Port: 6175

Instance Name: PPA

Frame Rate: 30

Metadata Timeout: 60000

Start Time: 06/27/2017 15:55:00.000

End Time: 06/27/2017 15:56:00.000

Point List: FILTER MeasurementDetail WHERE SignalAcronym IN (\'VPHM\', \'IPHM\', \'VPHA\', \'IPHA\')

Message Interval: 2000

Enable Logging: True

[The Point List can also be indicated as Filter Expression, relevant documents can be found on page: <https://github.com/GridProtectionAlliance/openPDC/blob/master/Source/Documentation/wiki/Connection_Strings.md#syntax-for-inputmeasurementkeys-and-outputmeasurements> ]

1. The result of the application is a data set stored in the application, which can be taken as an input for other applications. The structure of the data set is shown as following:



