Kepware Technologies

Benchmarking the Performance of KEPServerEX Projects

Table of Contents

1. Overview

Members of Kepware's Technical Support team are frequently asked about methods for improving the rate at which data is acquired from a device or devices in a process. The answer is not simple: there are many parameters that can impact performance. The list includes (but is not limited to) the following:

- The type of media connection (serial, Ethernet, and so forth).

- The bandwidth of the connection.

- The number of devices dropped off the media connection.

- The length of time it takes the device to process its ladder and respond to data requests.

- The number of connections to the device.

- The amount of data being requested.

- The rate at which data is requested.

**Note:** For more information on optimizing KEPServerEX, refer to the server help file.

Before attempting to improve performance, users should establish a baseline. This document intends to show users how to utilize KEPServerEX's built-in diagnostics to benchmark the communications process between the server and devices.
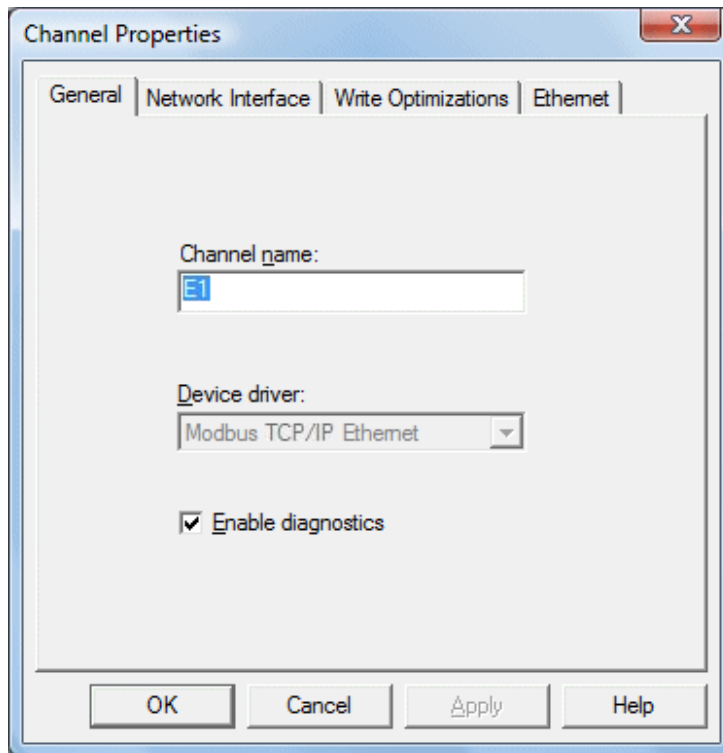
2. Benchmarking Project Performance

2.1 Preparing the Server Project

Channel Diagnostics must be enabled in order to capture benchmark data.

1. To start, double-click on the channel to open Channel Properties. In this example, the channel is named "E1".
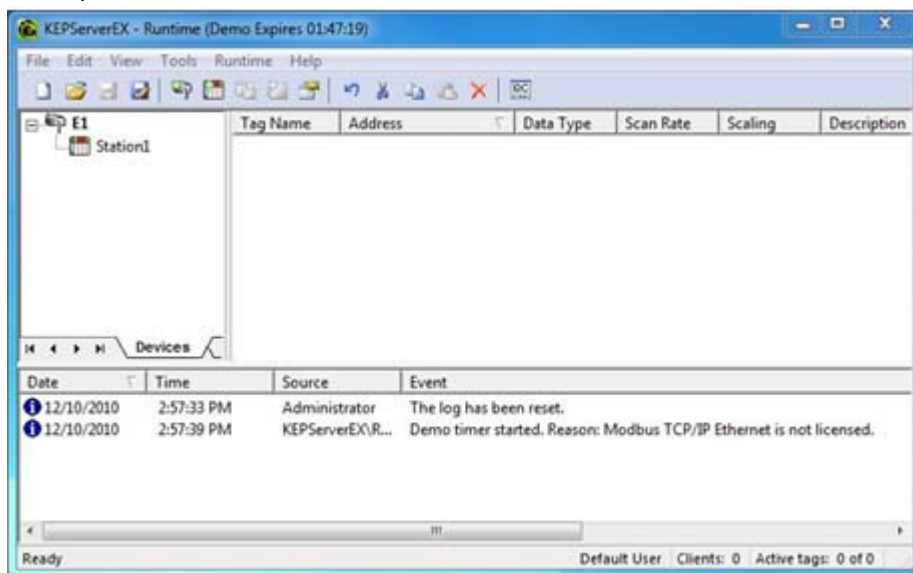
2. In the **General** tab, check the **Enable Diagnostics** checkbox.

3. Then, click **OK**.

4. On the server's main menu bar, click **View**. Scroll to **Diagnostics**, and then select **Channel**.

5. At this point, nothing else needs to be done in the server.

2.2 Configuring the OPC Quick Client to Request Data

In order to sample a set of data, the client must request that data from the server. Most clients are configured to continually update data by periodically issuing new data requests to the server. For this benchmark measurement, clients must be configured to only request data once. A single request for a set of data usually consists of multiple read requests and associated read responses. This benchmark requires that the time from the first read request to the time of the last read response be determined. In the following example, OPC Quick Client will be used.
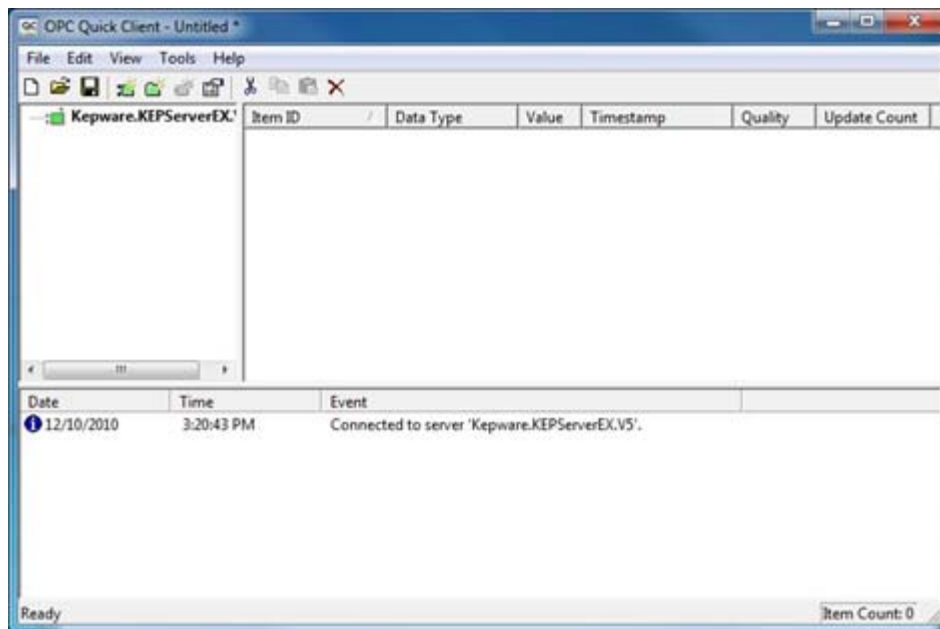
1. To start, manually launch the OPC Quick Client by selecting **Start** | **Kepware** | **KEPServerEX 5** | **OPC Quick Client**.
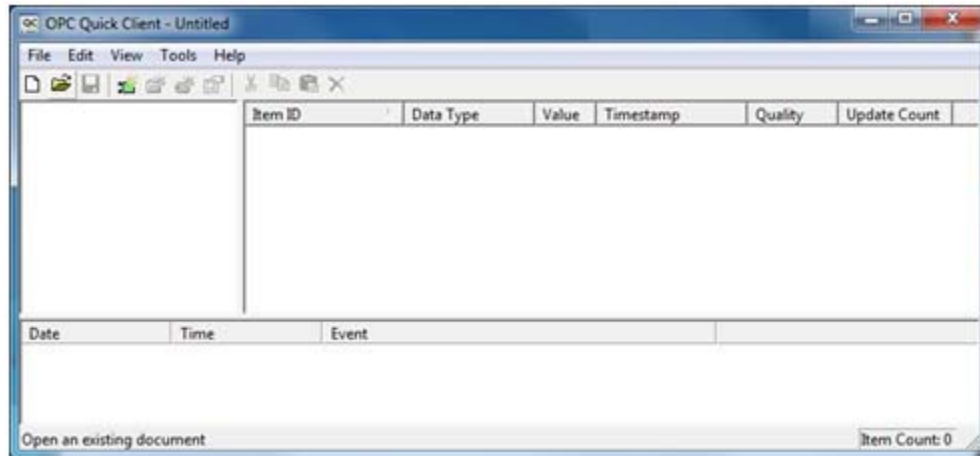

2. Next, add a new server connection by clicking the **New Server** icon .
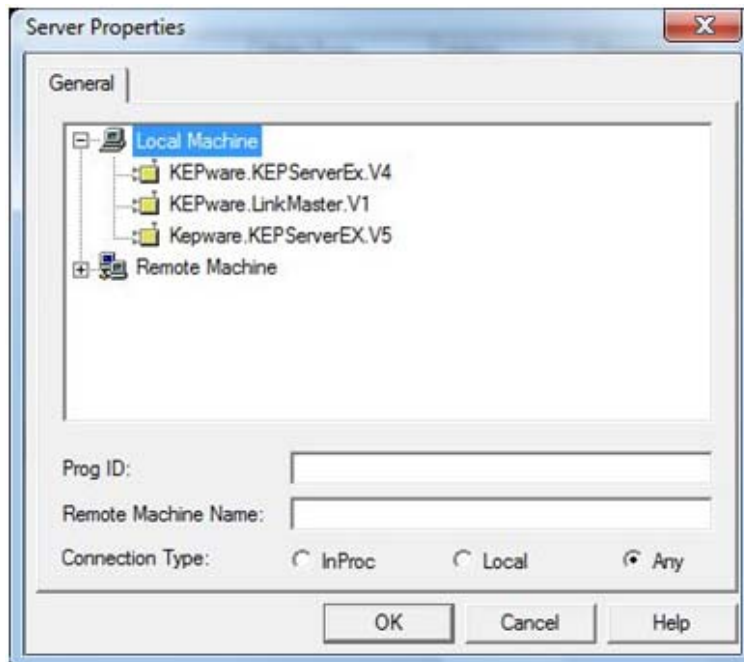
3. Locate the server from which the benchmark data will be captured and then click **OK**. In this example, the server selected is "Kepware.KEPServerEX.V5".
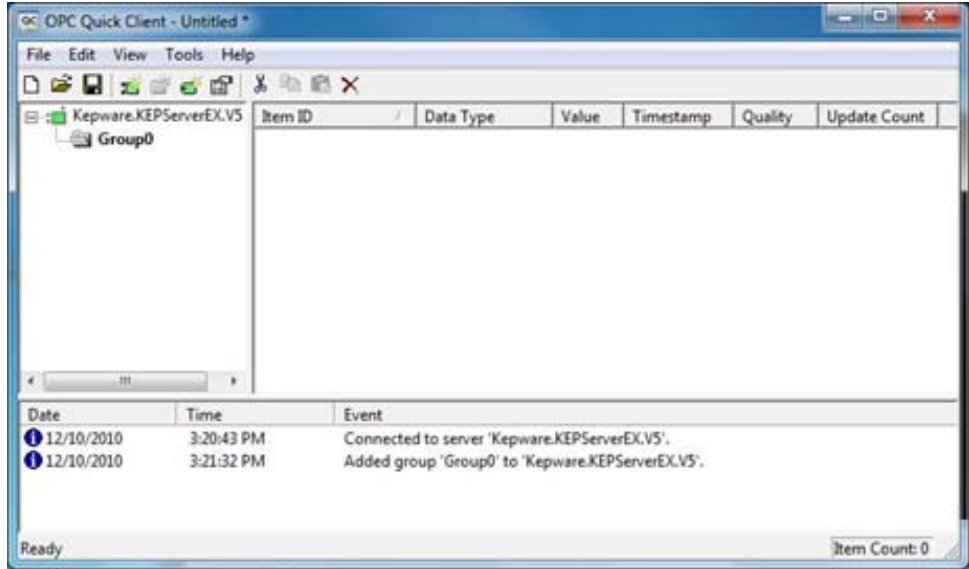
4. Next, add a new OPC Group by clicking the **New Group** icon .

5. In **Group Properties,** uncheck **Active State**. This will prevent the client from polling added OPC items.
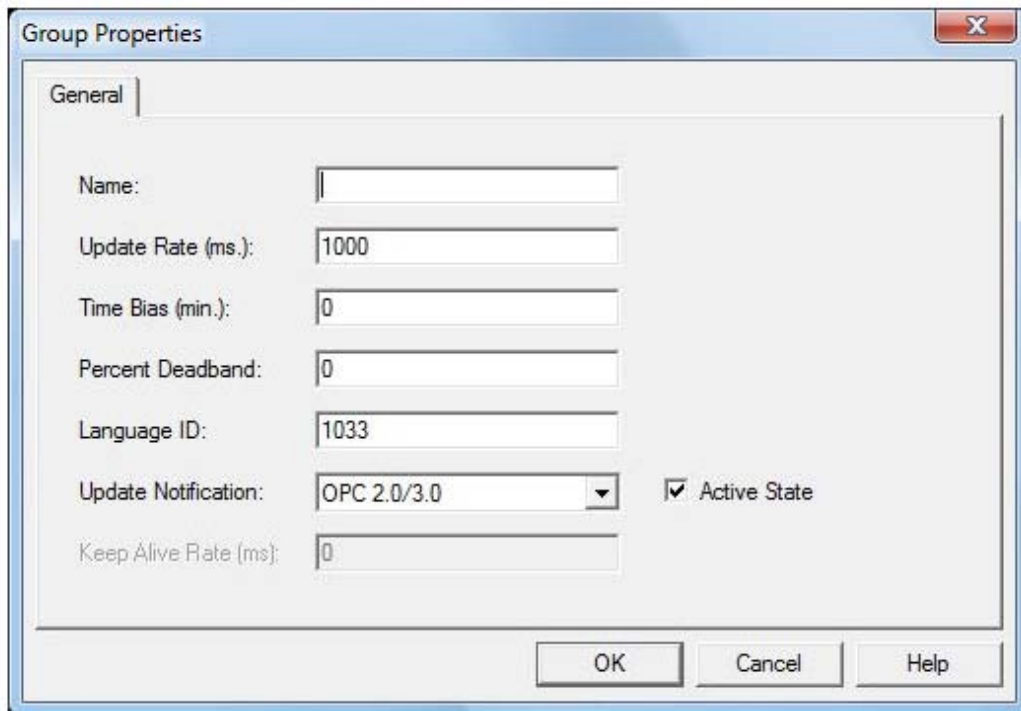
6. Leave the **Name** field blank, and then click **OK**.

Note: "Group0" should now be visible in the OPC Quick Client project tree.

7. Next, add items. To do so, select "Group0" and then click the **New Item** icon .

8. In **Add Items**, locate the **Browsing** section. Then, expand the server connection tree.

---
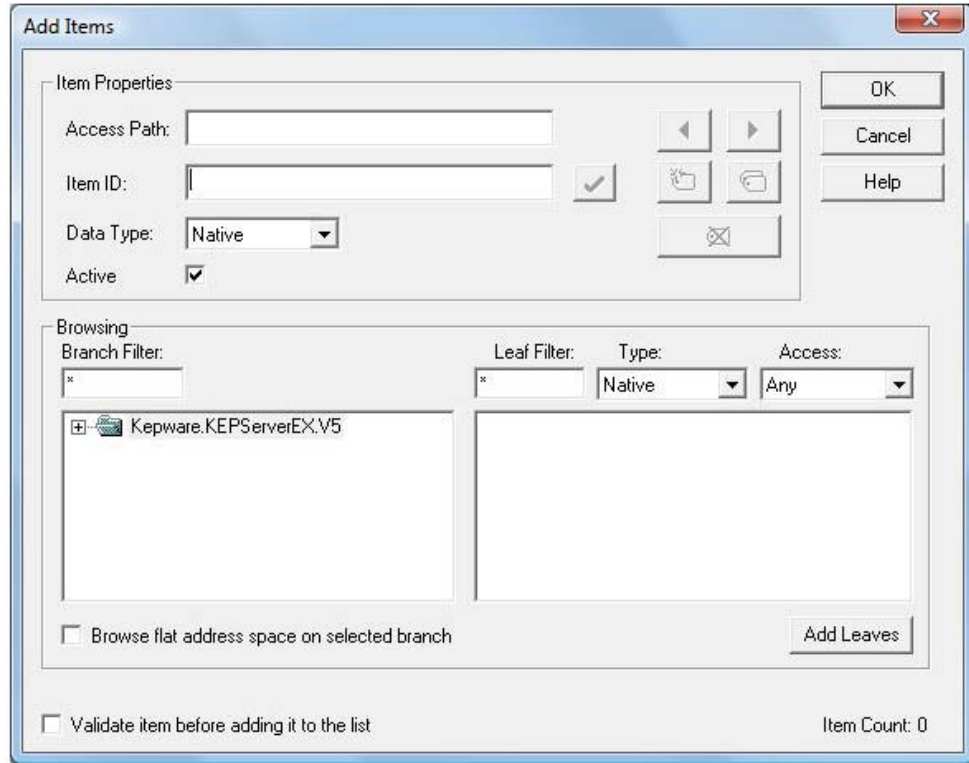
9. Locate the channel, and then expand it to locate and select the device. In this example, the device is named "Station1".

10. Next, select all of the tags.

11. To add the items to "Group0," click **Add Leaves**.

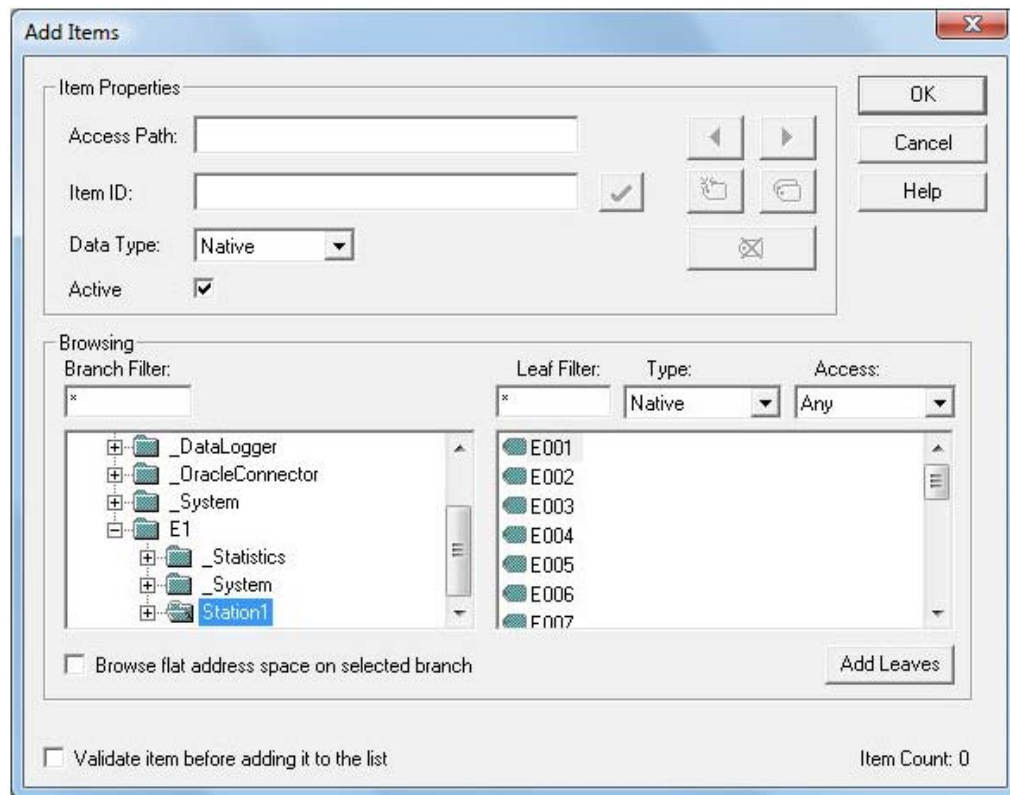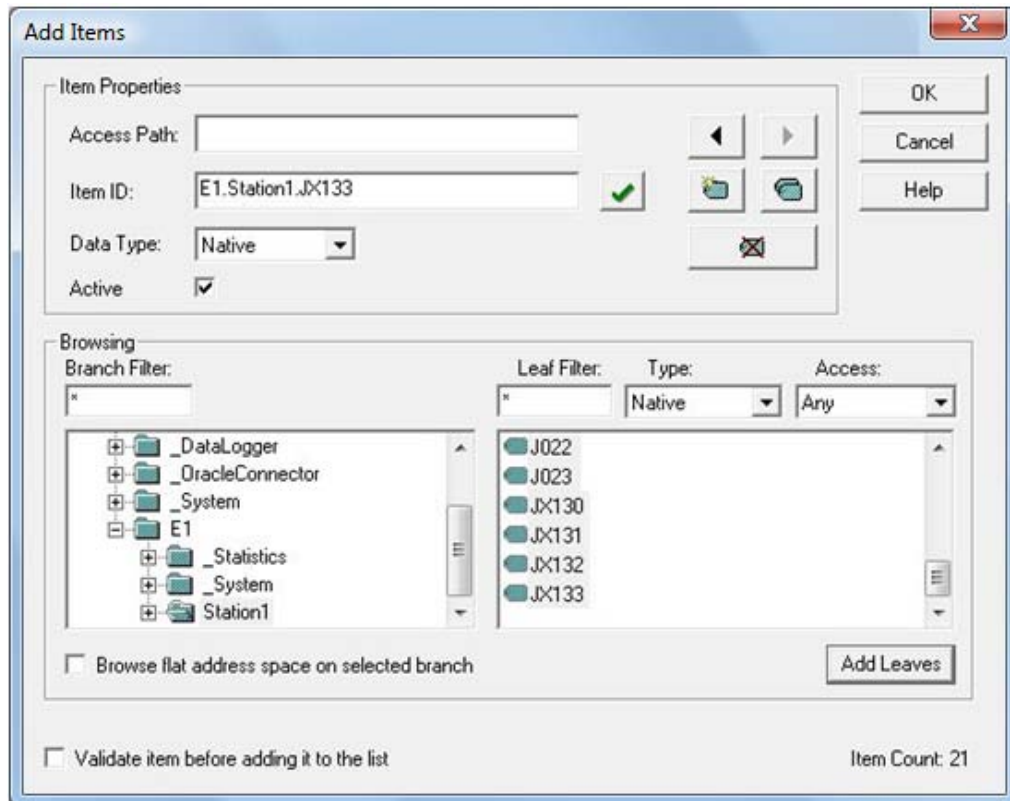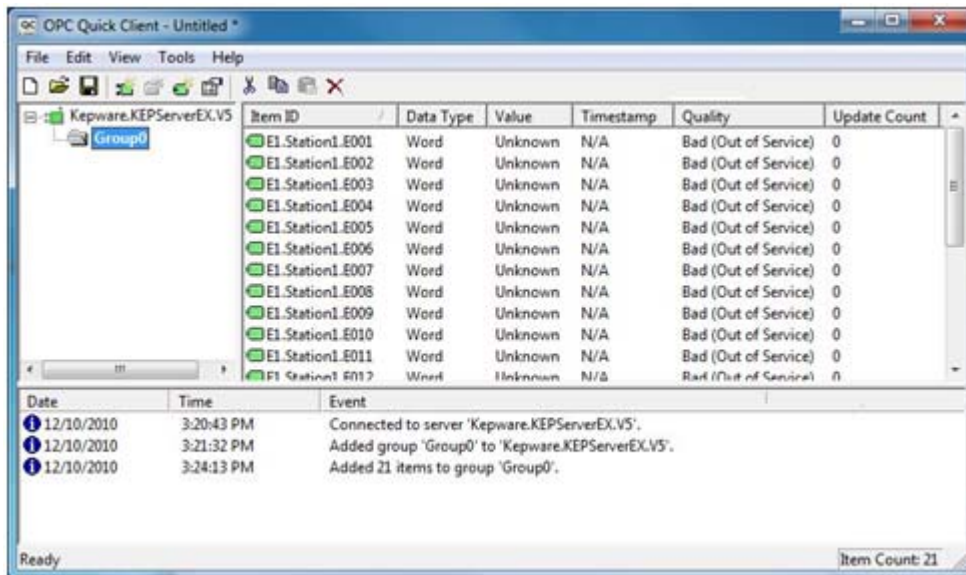12. Repeat the process until all tags that will be read have been added to the project. Then, click **OK**.

13. In the OPC Quick Client, select all the tags in "Group0". Then, right-click and select **Synchronous Device Read**.

2.3 Extracting the Benchmark Data from Channel Diagnostics

1. Next, return to the Channel Diagnostics view. All transmitted and received data exchanged in the Synchronous Device Read should be displayed.

   Note: The diagnostic View Panel displays both transmitted and received data. It also provides information on the total number of bytes transmitted, the total number of bytes received, and the total number of read and write requests.
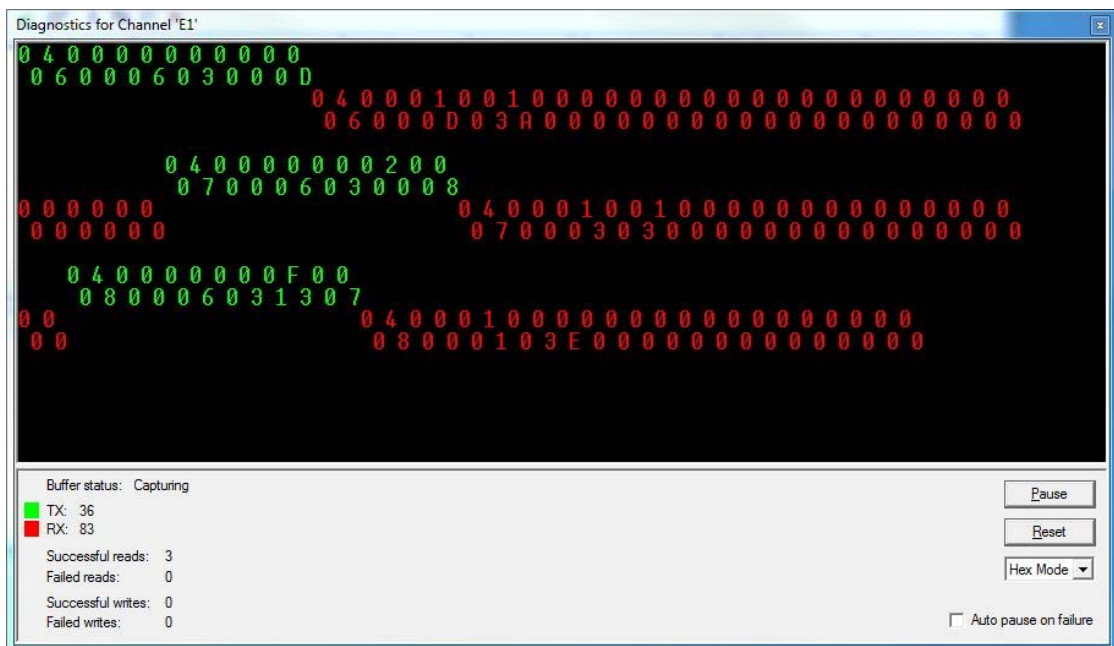
Note: The diagnostic View Panel color codes the data. Transmitted bytes default to dark green, and received bytes default to white. To change the background color, the text colors, the text font size, or set the capture buffer size, simply right-click in the View Panel to invoke the context menu.

2. To calculate the amount of time it took to collect the sample of data, right-click on the View Panel and then select **Copy to Clipboard**. Alternatively, select **Save as Text File** from the context menu. In this example, the diagnostics data has been pasted into **Notepad**.

```
12/13/2010  9:59:00.693 AM  TX: 00 01 00 00 00 06 00 03 00 00 00 0D
12/13/2010  9:59:00.794 AM  RX: 00 01 00 00 00 1D 00
12/13/2010  9:59:00.794 AM  RX: 03 1A 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00
12/13/2010  9:59:00.794 AM  TX: 00 02 00 00 00 06 00 03 00 20 00 0B
12/13/2010  9:59:00.794 AM  RX: 00 02 00 00 00 13 00
12/13/2010  9:59:00.794 AM  RX: 03 10 FF FF 00 4A FF FF 00 00 FF FF 00 00 FF FF 00 00
12/13/2010  9:59:00.894 AM  TX: 00 03 00 00 00 06 00 03 01 F3 00 08
12/13/2010  9:59:00.895 AM  RX: 00 03 00 00 00 13 00
```

3. The copied diagnostics will include the date and time for each transaction. To get the total amount of time required to complete a single read of all the data from the device, subtract the time of the first transmit (TX) from the time of the last received (RX) packet. In this example, it took 202 milliseconds.

Note 1: The Diagnostics buffer is limited. Users should beware of transaction packets that may be missing from the beginning of the diagnostics text file. This could throw off the time measurement considerably.

Note 2: Users should make sure that the diagnostic's buffer has been maximized. This can be done by right-clicking on the View Panel and then selecting **Options** | **Buffer Settings**. It may also be necessary to experiment with reads of different sizes to determine whether the buffer's limit has been exceeded. Using a Third-Party application in place of Channel Diagnostics (such as Wireshark) will avoid these buffer limitations.

2.4 Multiple Devices per Channel

When the project has multiple devices on a single channel, the items from the devices will be read in one request, providing the amount of time it would take to read the data from all devices once. When multiple devices are placed on a single channel, the server will read the data from each device in succession.

3. Summary

These instructions can be applied to any KEPServerEX driver that supports channel diagnostics. For more information on such drivers (or on the server in general), contact a Kepware sales representative.