



## Application Tutorial: Interfacing with a TCP/IP Weigh Scale

This exercise will walk you through retrieving information from a TCP/IP Weigh Scale. If you have completed the *Ethernet/Serial DataLink: Sending and Receiving Data over Ethernet or a Serial Port* tutorial, you are familiar with receiving simple data from this scale. In this exercise, you will use the Ethernet/Serial DataLink and an Input Command with a parsing script to retrieve all of the statistics in one Command execution.

### Requirements

In order to complete this exercise you must have the following:

- PC with Windows 2000 Professional or Windows 2000 Server
- I/Gear v5.2 or greater Installed
- Weigh Scale application
- An Ethernet Card for your PC

### Prerequisites

- Ethernet/Serial DataLink Tutorial: Sending and Receiving Data over Ethernet or a Serial Port

### Goals

By the end of this exercise, you will know how to:

- Interface with a Weigh Scale

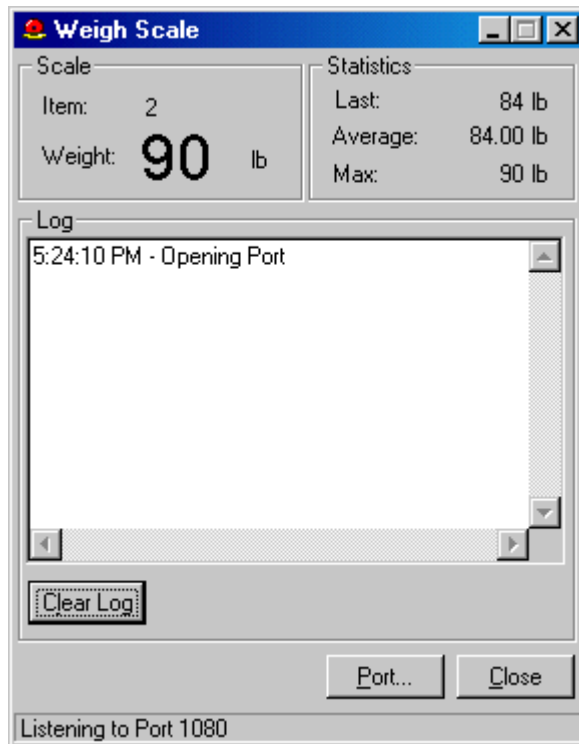
## Step 1 - Connecting to the Weigh Scale with the TCP/Serial DataLink

---

You will now configure the DataLink to connect to the Weigh Scale. This DataLink will provide a connection to Port 1080 on the local machine.

**Note:** If you have already completed the *Ethernet /Serial DataLink: Sending and Receiving Data over Ethernet or a Serial Port* tutorial, you have already created the **Weigh Scale** DataLink and you may skip to Step 2.

For this tutorial, you will use the Weigh Scale application on the local machine. The Weigh Scale application is an TCP/IP Emulator simulating an industrial weighing device. From this device, you can request certain weight values and statistics using the protocol outlined below. The DataLink will be configured to send and receive data on Port 1080. The Weigh Scale application, shown below, will indicate when data has been received and sent through the port. The application is located in the Start menu at **Programs->I-Gear->Tutorial->Weigh Scale**.



### How the Weigh Scale Works

Upon opening the Weigh Scale application, the scale will immediately begin “weighing” items. The current item is shown in the top left corner in the **Scale** section. The current weight being read by the scale is shown in large digits in the top left corner as well. Once this weight stays steady for a solid 5 seconds, the scale will reset and the next item will be placed on the scale. The right-hand corner shows current statistics for all items measured. This includes the weight of the last item, the average weight of all the items, and the maximum weight reached.

The **Log** window shows the current happenings for the TCP/IP Port of the scale. On startup of the application, the port is opened and the application listens for incoming connection requests, which in this case, will come from I/Gear’s Ethernet /Serial DataLink. The port is defaulted to 1080, but may be changed by using the Port... button. Once a connection is established, the scale reads in the command request, one of the protocol commands shown below, and then returns the requested data. The connection to I/Gear is closed and the scale begins listening for another connection request.

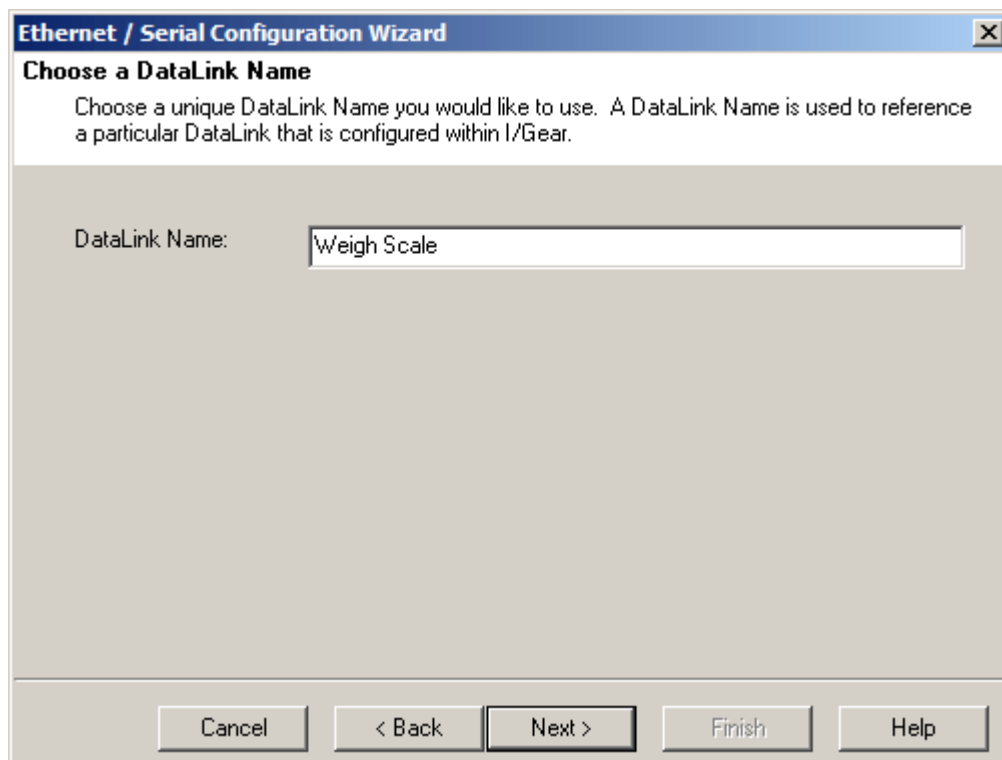
## Weigh Scale Communication Protocol

The protocol for the Weigh Scale is used to retrieve statistics and weight values from the device. All commands must be sent with a carriage-return line feed as a data packet terminator. To do this in I/Gear, you specify a control character for Carriage-return and Line Feed as outlined later in this tutorial. Here are the commands utilized by the Weigh Scale.

Command	Description
<b>/M</b>	Returns the maximum weight
<b>/A</b>	Returns the average weight
<b>/C</b>	Returns the current weight
<b>/L</b>	Returns the weight of the last item.

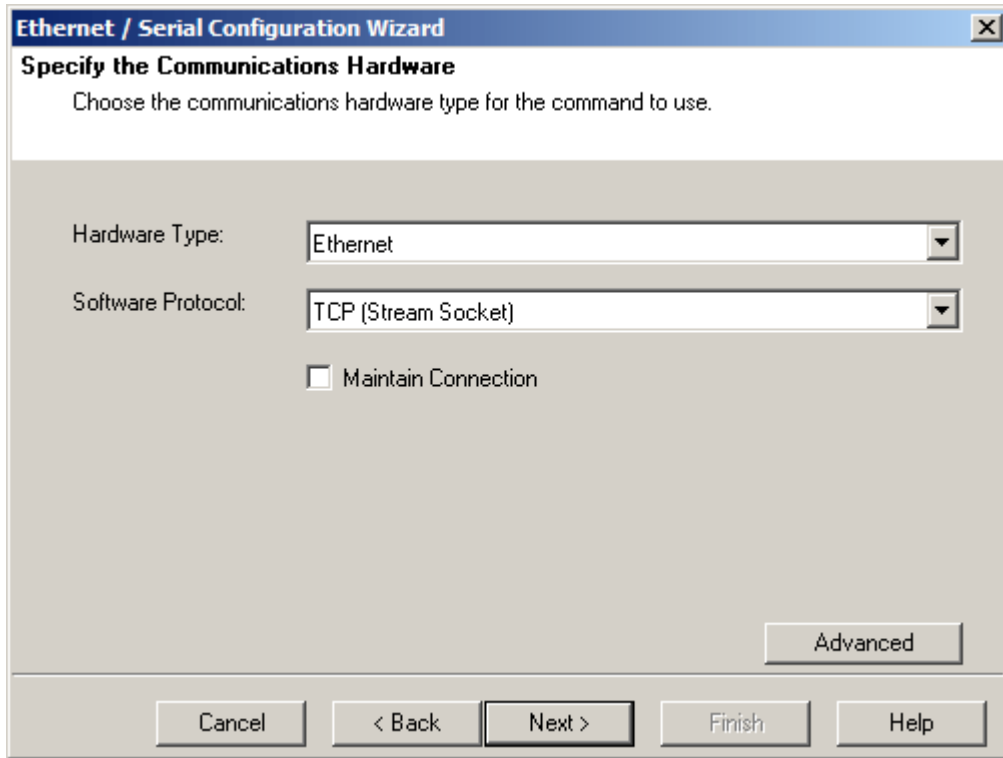
You will now create the Ethernet /Serial DataLink.

1. Right-click the **DataLinks** node in the tree.
2. On the pop-up menu, select **New DataLink...**
3. Select **Ethernet /Serial** from the list of DataLinks.
4. When the Configuration Wizard opens, click Next.
5. On the **Choose a DataLink Name** screen, type **Weigh Scale** and click Next.

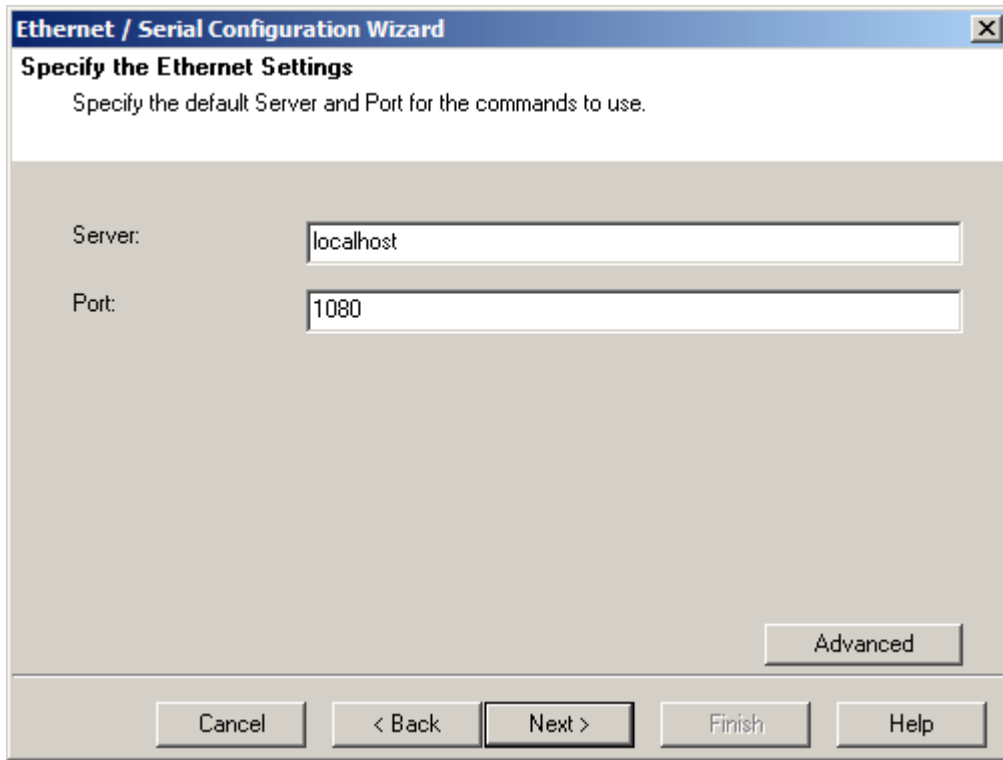


6. On the **Specify the Communications Hardware** screen, select the hardware for communication. There are two available communication hardware types:
  - **Ethernet** – communication over Ethernet.  
This option has four related Software Protocol options: TCP (Stream Socket), UDP (Datagram), ICMP (Ping), and Telnet.
  - **Serial Port** – communication over a Serial connection.

You will be using Ethernet to communicate with the TCP Server. Select **Ethernet**, **TCP (Stream Socket)**, leave **Maintain Connection** unchecked, and click Next.



7. On the **Specify the TCP Server** screen, enter **localhost** as the Server and **1080** as the Port and click Next. This will establish a connection on Port 1080 to the local computer. It is essentially a loop configuration where data will be sent and received on the local machine. In most cases, you will be connecting to a remote server or device using a name or IP Address.



8. Click Next on the **Specify any Protocol Modules** screen, and Click Finish on the **Completing the Ethernet/Serial DataLink Configuration Wizard** screen.

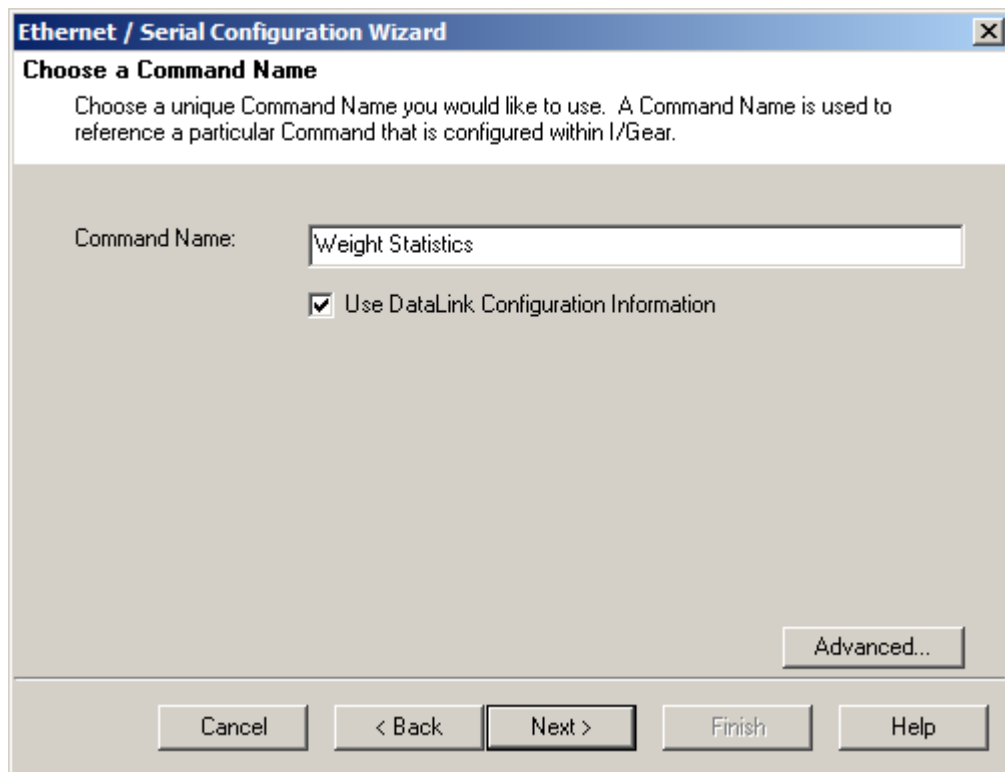
You have now configured the Ethernet /Serial DataLink for connecting to the TCP/IP MES Interface Portal that will be running on the local machine. The next step is to create an Input Command to interact with the MES system.

## Step 2 - Creating the Interactive TCP Input Command

---

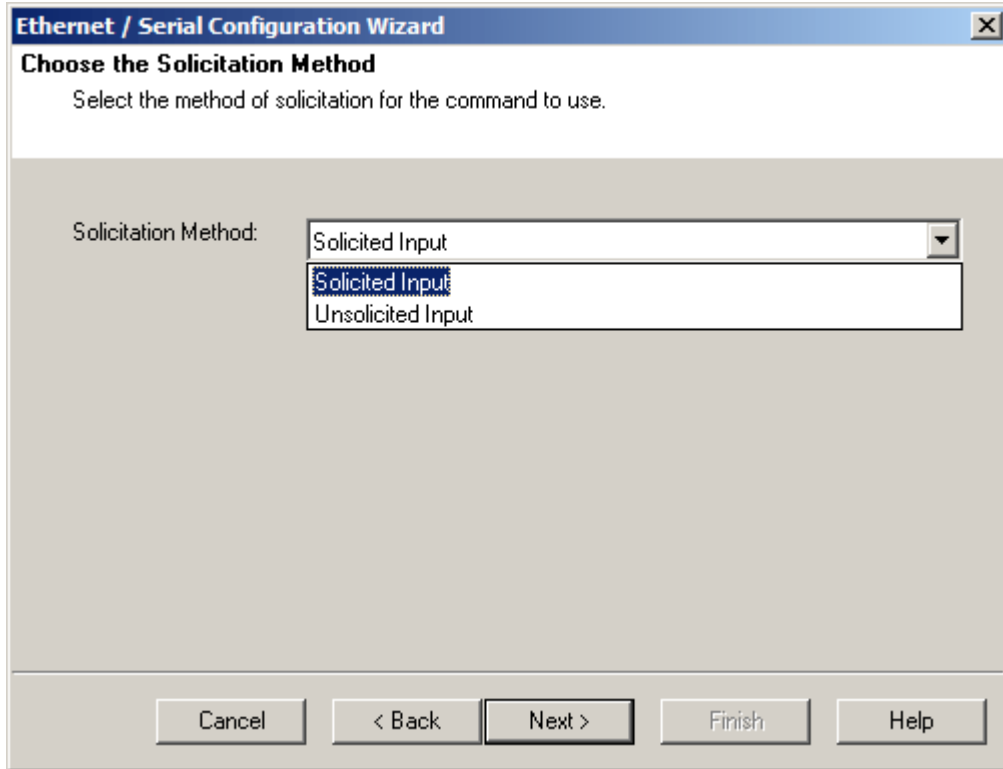
You will now create an Input Command that uses an Interactive Session with a Parsing Script to automatically respond and retrieve statistics from the Weigh Scale application.

1. Expand the **Weigh Scale** DataLink tree
2. Right-click the **Input Commands** node in the tree.
3. On the pop-up menu, select **New Command...**
4. When the Configuration Wizard opens, click Next.
5. On the **Choose a Command Name** screen, type **Weight Statistics** and click Next.



6. On the **Choose the Solicitation Method** screen, you will choose how to execute the Command. Unsolicited Commands are executed when data is received. Solicited Commands are executed when they are triggered.

Since you are setting this Command up to execute each time you trigger it, choose Solicited Input as shown below and click Next.



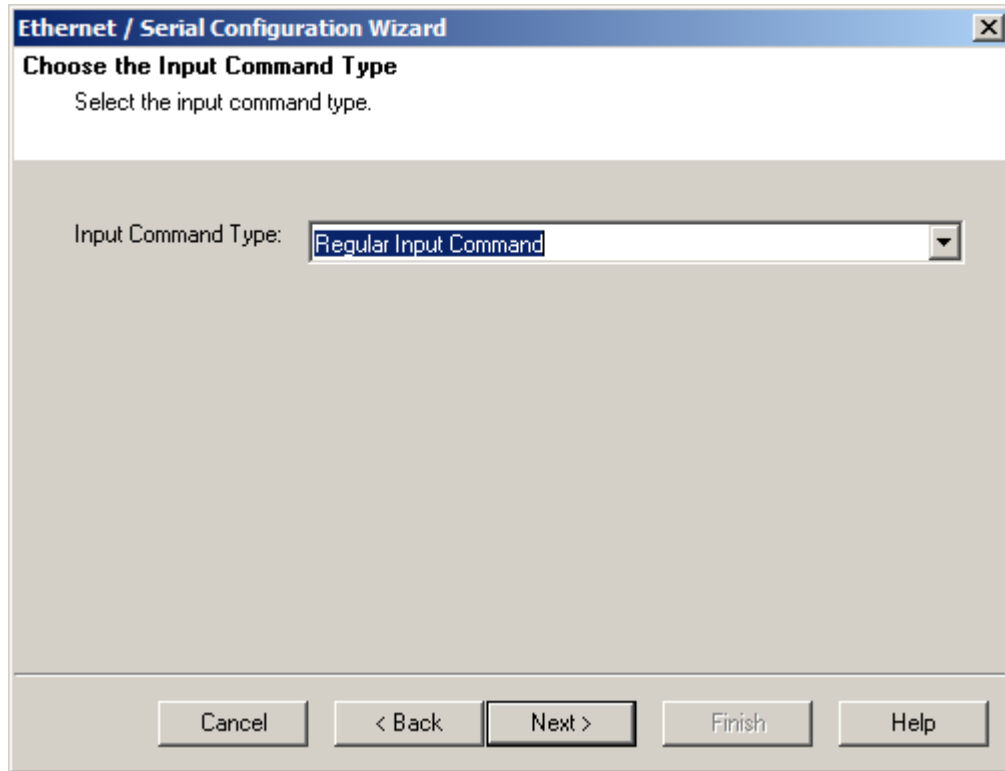
7. On the **Specify a Trigger Expression** screen, select a Trigger for the Command. There are several types of Triggers available:
- **Time Type** – Trigger occurs on a specified time interval from a specific base time such as every one second.
  - **Event Type** – Trigger occurs on certain events of an existing Command or DataPoint, such as on success of the Command or on increase of the DataPoint.
  - **Custom** – Specified by the user using specific syntax and Boolean rules.
  - **External** - The Command can only be executed from the I/Gear Object Model or through the Management Console.

You will be executing the Command from the Management Console. Therefore, select **External Triggering** and click Next.

The screenshot shows a dialog box titled "Ethernet / Serial Configuration Wizard" with a sub-header "Specify a Trigger expression". Below the sub-header is a descriptive text: "Specify the Trigger expression you would like to use. The Trigger expression is used by I/Gear to determine when to execute a particular command." There are three radio button options: "Basic Triggering", "Custom Triggering", and "External Triggering". The "External Triggering" option is selected. Under "Basic Triggering", there are two sub-sections: "Time Type" (checked) and "Event Type" (unchecked). The "Time Type" section includes a "Time Base" dropdown menu showing "Wed Apr 10, 2002 4:12:38 PM", a "Use Local Time" checkbox (checked), and an "Every" section with a text input "1" and a dropdown menu "Minute(s)". The "Event Type" section includes a "DataPoint" text input field with a browse button "...", and an "Action" dropdown menu. Under "Custom Triggering", there is a text input field containing "EXTERNAL". At the bottom of the dialog are five buttons: "Cancel", "< Back", "Next >", "Finish", and "Help".

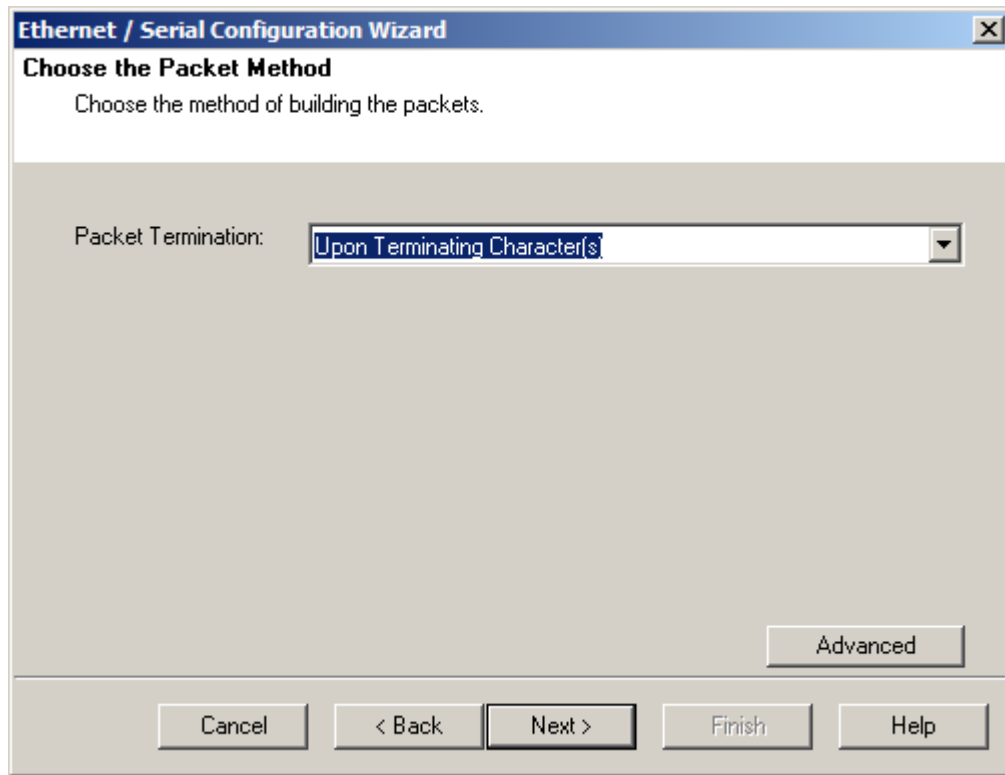
8. On the **Choose the Input Command Type** screen, you can choose the type of Input Command. The available types are as follows:
- **Regular Input Command** allows the Command to receive and send data through the device.
  - **Connection Statistics Command** gives diagnostic information about the device.

Since you are creating this Command to receive data, select **Regular Input Command** and click Next.

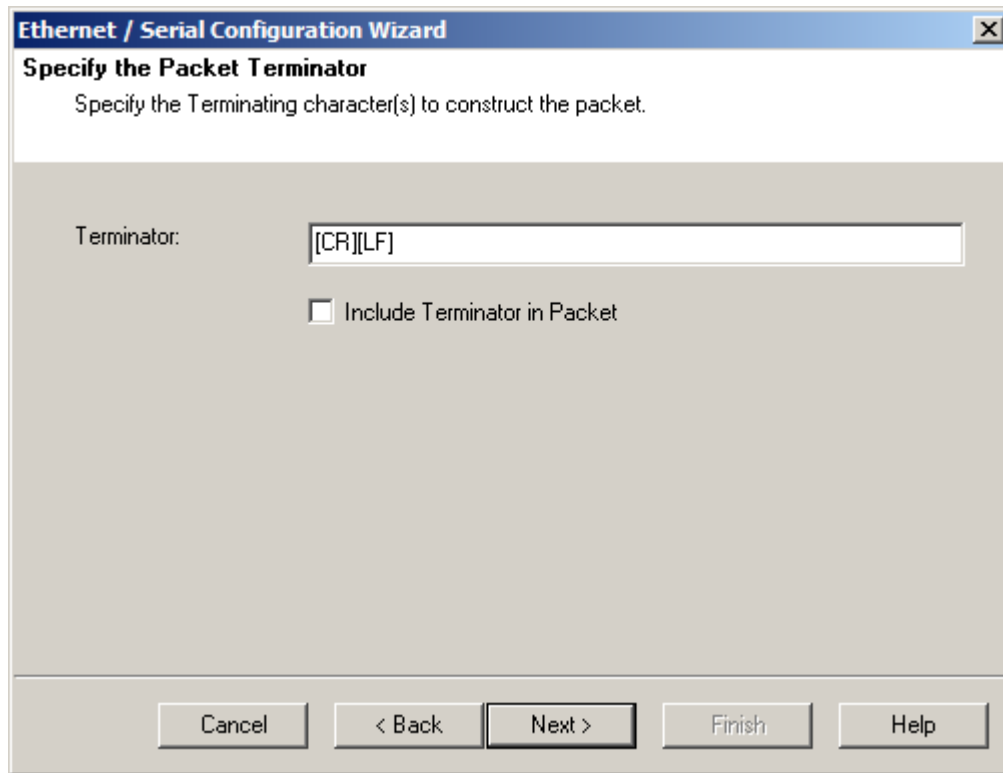


9. On the **Choose the Packet Method** screen, you will select the method of building packets. The available methods are as follows:
- **Upon Terminating Characters** will end the packet after the specified character(s) is received.
  - **Fixed Packet Length** will end the packet after a specified number of characters.
  - **Transmission Deadband** will end the packet after a specified amount of time passes with no data.

The Weigh Scale sends data with special characters on the end to signify the end of a packet. Therefore, select **Upon Terminating Character(s)** and click Next.



10. On the **Specify the Packet Terminator** screen, verify **[CR][LF]** is entered as shown and that the **Include Terminator in Packet** option is unchecked. This will indicate that the incoming part information is complete once a carriage-return line feed is reached. This will be specific to the device that is sending data back. The Weigh Scale application sends the data with a carriage-return line feed to indicate the packet has ended.



By default, the brackets ([]) indicate a control character, telling I/Gear that these are special characters that need to be interpreted as control characters instead of as part of the data. The start and end control characters can be changed in the Advanced Properties of the Ethernet /Serial DataLink. The chart below outlines the characters that can be used. These characters can also be used in transmitted and incoming data packets as well.

DEC	HEX	ASCII
0	00	NUL
1	01	SOH
2	02	STX
3	03	ETX
4	04	EOT
5	05	ENQ
6	06	ACK
7	07	BEL
8	08	BS
9	09	HT
10	0A	LF
11	0B	VT
12	0C	FF
13	0D	CR
14	0E	SO
15	0F	SI

DEC	HEX	ASCII
16	10	DLE
17	11	DC1
18	12	DC2
19	13	DC3
20	14	DC4
21	15	NAK
22	16	SYN
23	17	ETB
24	18	CAN
25	19	EM
26	1A	SUB
27	1B	ESC
28	1C	FS
29	1D	GS
30	1E	RS
31	1F	US

For another example, if you wanted the packet to end on ETX, you could specify [ETX] in character notation or [x03] in hexadecimal notation.

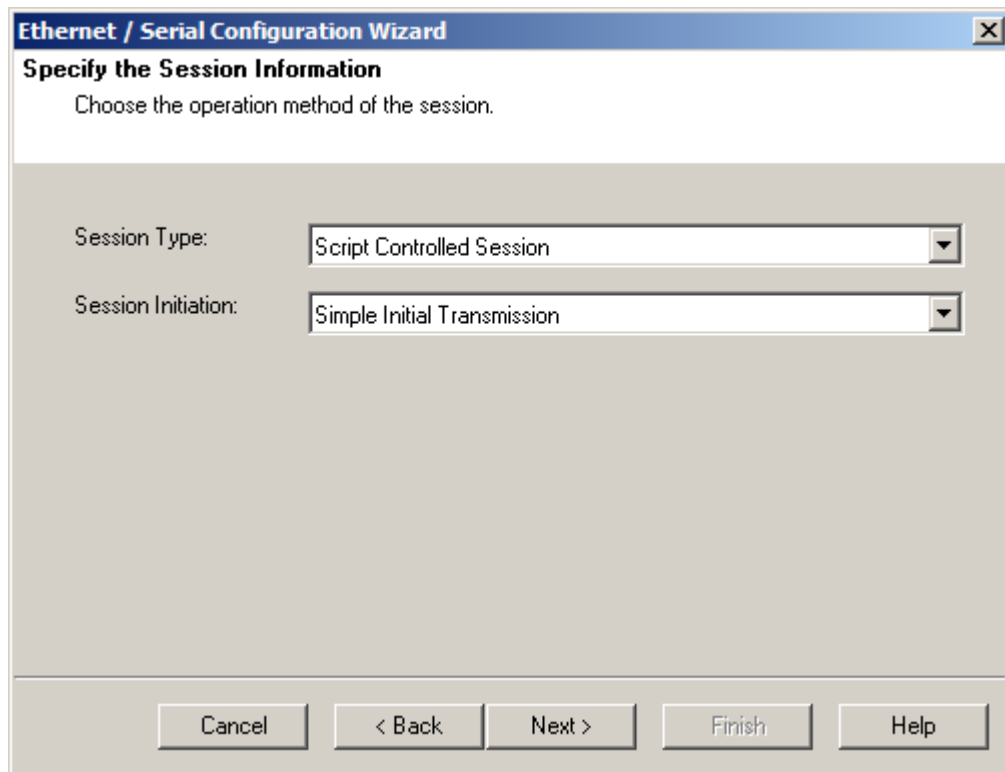
11. On the **Specify the Session Information** screen, you can select a session type and session initiation. There are two available session types:

- **Standard Session** is a basic session type that doesn't need any scripting.
- **Script Controlled Session** adds scripting to the session.

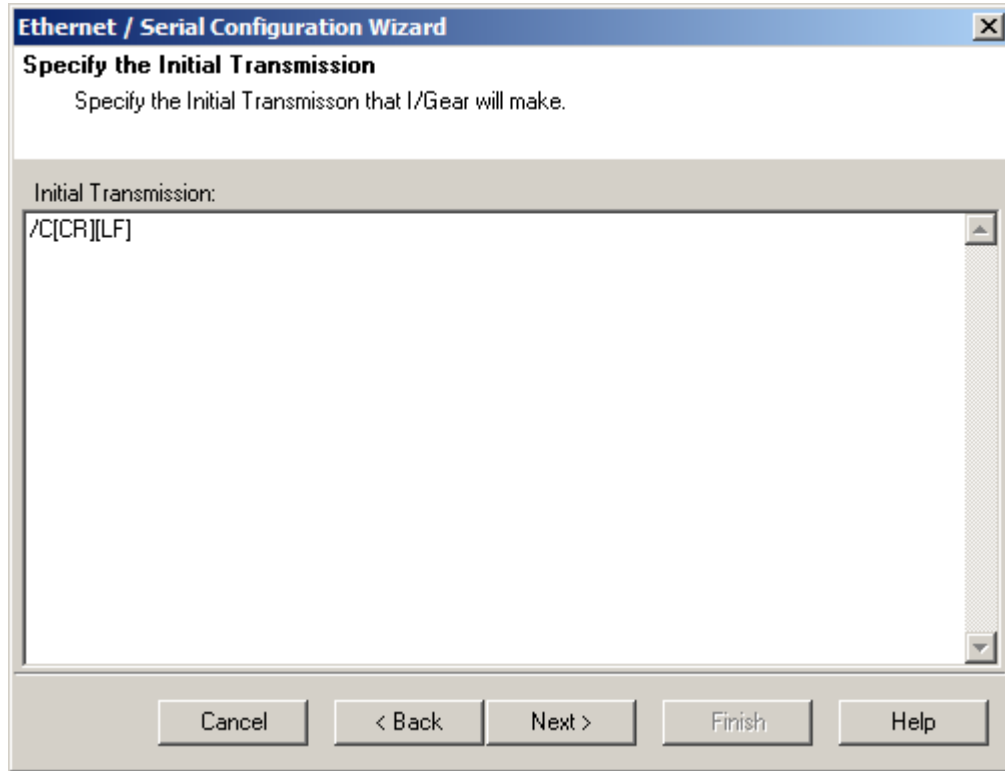
The Session Initiation also has two options:

- a. **No Initial Transmission** is used when the data is returned right after the connection.
- b. **Simple Initial Transmission** is used when the data is returned only after a request message is sent.
- c. **Script Controlled Initial TX** (only available with **Script Controlled Session**) allows the initial transmission to be set up in a script.

Since you will be reading data from the device and parsing it with script, select **Script Controlled Session** and **Simple Initial Transmission** as shown below and click Next.



12. On the **Specify the Initial Transmission** screen, enter `/C[CR][LF]`. Some devices or servers require a command or request upon connection indicating what data or operation you are requesting from them. This is part of their communication protocol and is required before you can exchange data with them. In this case, `/C` tells the TCP Server that you wish to obtain the current weight of the scale first during this session as noted by the protocol described at the beginning of this exercise. The Weigh Scale also terminates incoming data packets with carriage-return line feeds, hence the ending `[CR][LF]`. When finished, click Next.



13. On the **Specify the Parsing Script** screen, you will enter a Parsing Script. A Parsing Script is a VBScript macro that is used to parse information, transmit response to data received, and control the exchange of data with the TCP Server, the Weigh Scale. Enter the Parsing Script shown below. This script will execute every time data is received by the Command.

```
'Define any Global Scope Variables here
Public Current_Weight
Public Last_Weight
Public Maximum_Weight
Public Average_Weight

Public Sub Parse()
  Dim sTrans

  sTrans=Command.LastTransmission

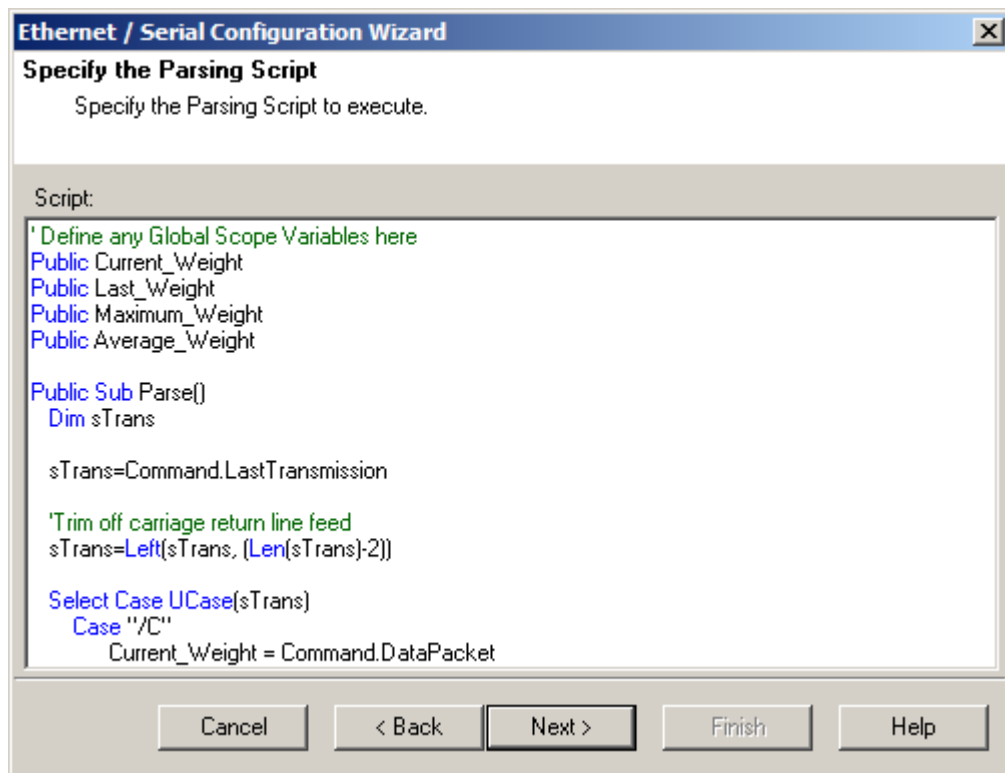
  'Trim off carriage return line feed
  sTrans=Left(sTrans, (Len(sTrans)-2))

  Select Case UCase(sTrans)
    Case "/C"
      Current_Weight = Command.DataPacket
      Command.Transmit "/L[CR][LF]",TRUE
    Case "/L"
      Last_Weight = Command.DataPacket
      Command.Transmit "/M[CR][LF]",TRUE
    Case "/M"
      Maximum_Weight = Command.DataPacket
      Command.Transmit "/A[CR][LF]",TRUE
    Case "/A"
      Average_Weight = Command.DataPacket
      Command.Complete TRUE, ""
  End Select
End Sub
```

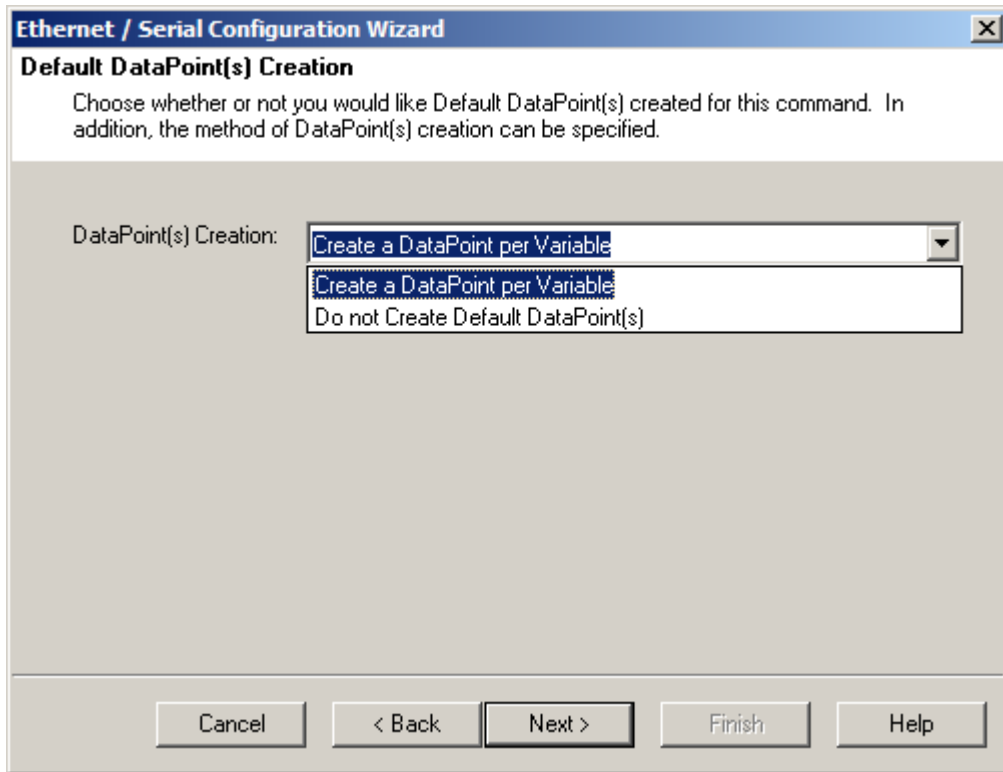
Any global variables can be accessed as DataPoints. The wizard will give you the option of creating DataPoints for the variables automatically, or you can create them later through the Management Console. In this example, there are four global variables that will be used to hold the incoming data – **Current\_Weight**, **Last\_Weight**, **Maximum\_Weight**, and **Average\_Weight**. These will become the DataPoints for the Command.

The **Command Object** is used to access properties and methods available for control and information of the current Command. The following properties and methods are available in the Parsing Script.

- **Command.PacketNumber** – returns the 1-based packet sequence for the Command.
- **Command.LastTransmission** – returns the last packet of data sent by the Command to the server.
- **Command.DataPacket** – returns the packet of data that has been received by the Command.
- **Command.Transmit(sTX, bReplace)** – transmits data to the server.
  - **sTX** – the packet of data to be transmitted.
  - **bReplace** – whether or not the control codes sequences should be replaced in the sTX string.
- **Command.Complete(bStatus, sMessage)**
  - **bStatus** – set to True if Command is successful, or False if it fails.
  - **SMessage** – string indicating the state of the Command on completion.

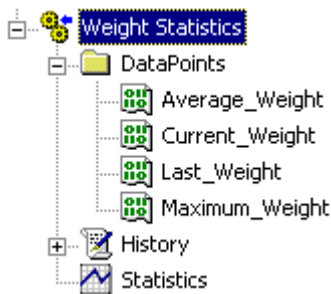


14. On the **Default DataPoint(s) Creation** screen, select **Create a DataPoint per Variable**. This will create a DataPoint for each Global Variable of the Script – Current\_Weight, Last\_Weight, Maximum\_Weight, and Average\_Weight. When finished, click Next.



15. Click **Finish** on the **Completing the Ethernet /Serial Command Configuration Wizard** screen.

The **Weight Statistics** Command should now be in the Component Tree. It should contain four DataPoints as shown below. These DataPoints represent the four global variables created at the top of the script.

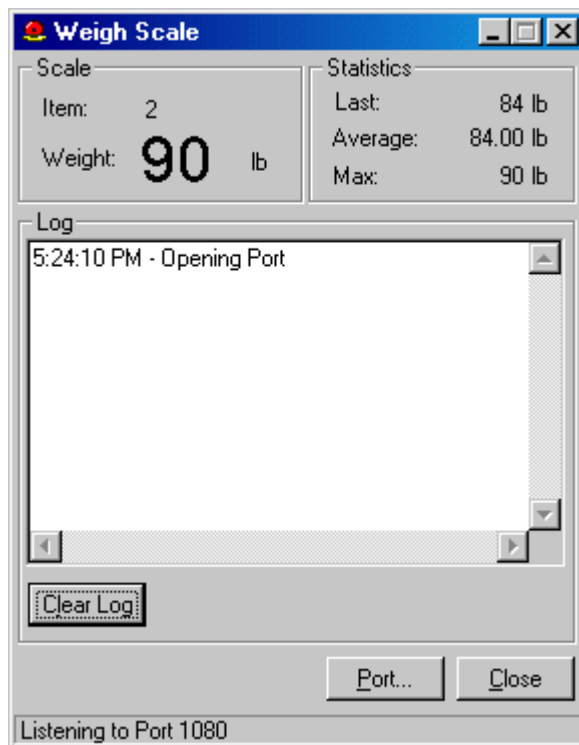


### Step 3 - Let's Go!

---

This step will guide you through opening the Weigh Scale application and instruct you on what is happening when the Command executes to further your understanding of the interactive capabilities of the TCP/Serial DataLink.

1. Open the Weigh Scale application by going to **Programs->I-Gear->Tutorial->Weigh Scale** in the Windows Start menu. When the scale opens, the application will automatically be weighing items and will be listening to port 1080 for incoming connection requests.



2. You will now execute the **Weight Statistics** Command by right-clicking it in the Component Tree and selecting **Execute** from the pop-up menu. Upon the first execution, the Command will establish a connection with the Weigh Scale and send the initial transmission string **/C**.
3. Once the Weigh Scale application receives the initial transmission string, the current weight will be sent back to I/Gear.
4. Upon receipt of the data, the **Weight Statistic** Command will fire the Parsing Script. Its first action is to set a local variable, *sTrans*, to the **Command.LastTransmission**. All transmissions from the Command contain a [CR][LF] termination. Therefore, it is necessary to trim the last two characters of the transmission string off. This process is indicated by following code segment.

```
Dim sTrans
sTrans=Command.LastTransmission
'Trim off carriage return line feed
sTrans=Left(sTrans, (Len(sTrans)-2))
```

- At this point, *sTrans* will be /C. The **Select Case Ucase(sTrans)** performs a comparison of an uppercased *sTrans* with the given **Case** statements. It will evaluate the Case statement shown below as True, store the incoming data in the Current\_Weight variable, and transmit the next Command, /L, back to the Weigh Scale application to request the Last weight.

```
Case "/C"  
  Current_Weight = Command.DataPacket  
  Command.Transmit "/L[CR][LF]", TRUE
```

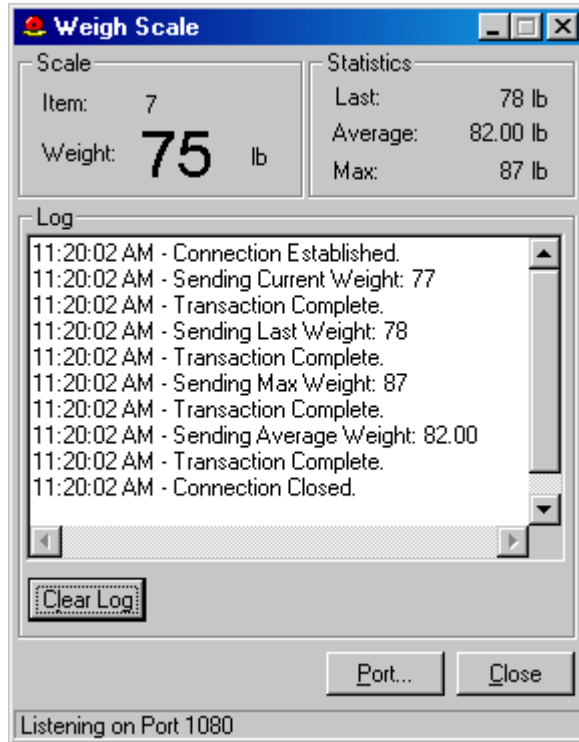
- Once the Weigh Scale receives the /L Command, it will send back the Last weight. The Command.LastTransmission is /L. Therefore, the Case statement shown below is true and the Command will save the incoming Data in the Last\_Weight variable and transmit the next requested data command, /M.

```
Case "/L"  
  Last_Weight = Command.DataPacket  
  Command.Transmit "/M[CR][LF]", TRUE
```

- The Command will continue to execute receiving and storing the Maximum and Average weights. Once the Average is received, the Command will be completed successfully.

```
Case "/M"  
  Maximum_Weight = Command.DataPacket  
  Command.Transmit "/A[CR][LF]", TRUE  
Case "/A"  
  Average_Weight = Command.DataPacket  
  Command.Complete TRUE, ""
```

- After the Command has been completed, the DataLink will close the connection with the Weigh Scale. The Log window of the Weigh Scale provides a log of all events happening since the Command began executing.



The values that were sent are indicated and should correspond to the data now found in each of the four DataPoints of the **Weight Statistics** Command. The DataPoints containing the statistical data from the Weigh Scale are shown below.

Name	Raw Value	Value
Average_Weight	82.00	82.00
Current_Weight	77	77
Last_Weight	78	78
Maximum_Weight	87	87

Congratulations! You have successfully completed the applied Ethernet /Serial DataLink exercise.