Scanning from the console is done similarly to scanning in a winforms app with a few exceptions:

- 1. There is no SelectSourceDialog, another method for selecting a scanner will need to be implemented
- 2. The program has to be prevented from continuing after acquire. This demo accomplishes this with the ModalAcquire property being set to true.

Below is the simplest version of console based scanning. To complete the demo, implement a way to select a device from the list of devices. (The simplest solution to this is "return devices[0]")

## **C**#

tatic void Main(string[] args) { Acquisition acquisition = new Acquisition(); AddEvents(acquisition); count = 0; DeviceCollection devices = acquisition.Devices; Device selected = SelectDevice(devices); selected.HideInterface = true; selected.ModalAcquire = true; Console.Out.WriteLine("---Beginning Scan---"); selected.Acquire(); Console.Out.WriteLine("---Ending Scan---\n Press Enter To Quit"); Console.In.Peek(); } private static void AddEvents(Acquisition acquisition) { acquisition.ImageAcquired += new ImageAcquiredEventHandler(acquisition\_ImageAcquired); acquisition.AcquireFinished +=new EventHandler(acquisition\_AcquireFinished); acquisition.AcquireCanceled += new EventHandler(acquisition\_AcquireCanceled); } static void acquisition\_AcquireCanceled(object sender, EventArgs e) { Console.Out.WriteLine("Acquisition Canceled"); } static void acquisition\_AcquireFinished(object sender, EventArgs e) { Console.Out.WriteLine("Acquisition Finished"); } static int count; static void acquisition\_ImageAcquired(object sender, AcquireEventArgs e) { string filename = "out"+ count++ +".tif"; e.Image.Save(filename,ImageFormat.Tiff); Console.Out.WriteLine("Frame " + count +" Acquired. Saved At: "+filename); } private static Device SelectDevice(DeviceCollection devices) { //TODO: Implement a Methodology for selecting a device }

## **VB.NET**

rivate Shared Sub Main(ByVal args As String()) Dim acquisition As New Acquisition() AddEvents(acquisition) count = 0 Dim devices As DeviceCollection = acquisition.Devices Dim selected As Device = SelectDevice(devices) selected.HideInterface = True selected.ModalAcquire = True Console.Out.WriteLine("---Beginning Scan---") selected.Acquire() Console.Out.WriteLine("---Ending Scan---" & vbLf & " Press Enter To Quit") Console.[In].Peek() End Sub Private Shared Sub AddEvents(ByVal acquisition As Acquisition) AddHandler acquisition.ImageAcquired, AddressOf acquisition\_ImageAcquired AddHandler acquisition.AcquireFinished, AddressOf acquisition\_AcquireFinished AddHandler acquisition.AcquireCanceled, AddressOf acquisition\_AcquireCanceled End Sub Private Shared Sub acquisition\_AcquireCanceled(ByVal sender As Object, ByVal e As EventArgs)

## HOWTO: Console Based Scanning in DotTwain

Console.Out.WriteLine("Acquisition Canceled") End Sub Private Shared Sub acquisition\_AcquireFinished(ByVal sender As Object, ByVal e As EventArgs) Console.Out.WriteLine("Acquisition Finished") End Sub Shared count As Integer Private Shared Sub acquisition\_ImageAcquired(ByVal sender As Object, ByVal e As AcquireEventArgs) Dim filename As String = "out" & count & ".tif" count +=1 e.Image.Save(filename, ImageFormat.Tiff) Console.Out.WriteLine(("Frame " & count & " Acquired. Saved At: ") + filename) End Sub Private Shared Function SelectDevice(ByVal devices As DeviceCollection) As Device 'TODO: CREATE METHODOLOGY FOR SELECTING DEVICE End Function

Original Article:

Q10275 - HOWTO: Console Based Scanning in DotTwain

Atalasoft Knowledge Base

https://www.atalasoft.com/kb2/KB/50196/HOWTO-Console-Based-Scanning-in-DotT...