Barcode Design In BarTender And Printing From LN

1. Problem Definition

a. Overview

The purpose of this document is explain the general process of designing and printing barcode labels including the software requirements, installation, export the design to a text file and use this file in LN or other application to print the labels. This process may change when software versions change.

Requirements

1. Computer with Windows 11.
2. Barcode printer, in this case we will use a Zebra printer for documentation purposes.
3. Bartender software: https://support.seagullsoftware.com/hc/en-us/articles/360058521853-Installing-BarTender?version=2022+and+later
4. Download Seagull barcode printer drivers for your printer(s): https://www.bartendersoftware.com/resources/printer-drivers
5. A Windows printing queue name to configure the printer in LN. \\ServerName\QueueName

b. Scope

Design a sample label, export the format to a text file and implement a way to print this file from LN using variable data.

2. Setting up the computer to do the design and export the label.

a. Software installation.

1- Install the full BarTender suite. The installation process is really straight forward so, just follow the instructions on the screen.
2- Install all the Seagull printer drivers in the development computer, even if the printer is a different model than the used in this document, the printer driver is used to export the barcode label to a text file.

b. Create Printer Device on Windows, General Instructions.

1- Goto Start Menu -> Devices and Printers.
   1. Click Add a printer.
   2. On the popup screen click Add a network, wireless or Bluetooh printer.
   3. Select Add a printer using a TCP/IP address or hostname, install on localhost. Click next
   4. Enter the printer ip address on ”Hostname or IP address:”. Enter port name. Do not check “Query the printer and automatically select the driver to use”. Click Next.
   5. Select the Device type. Standard→Select the Barcode Printer. Click Next.
   6. If Windows ask to share the printer, do not share it, just click Finish.

3. Designing and exporting the label.

a. Designing a new label.(to customize a label already created, just open the design).

1- Start BarTender suite, on the welcome screen, click close.

2- From the menu select File->New, the New Document Wizard starts

   1. Select Blank Template and click Next.

3- Select the barcode printer and click Next:

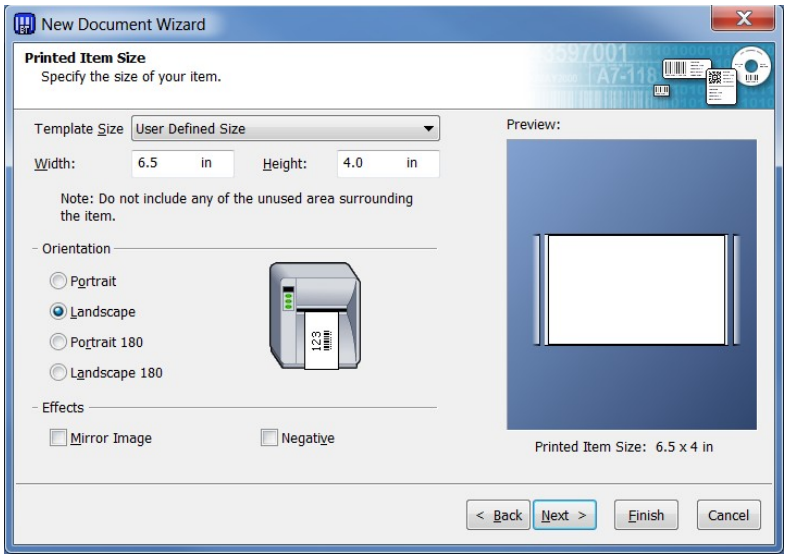
4- In the Stock Selection screen, select “Specify Custom Settings”. Click Next.

5- In Items Per Page, Select Single item per page. Click Next.

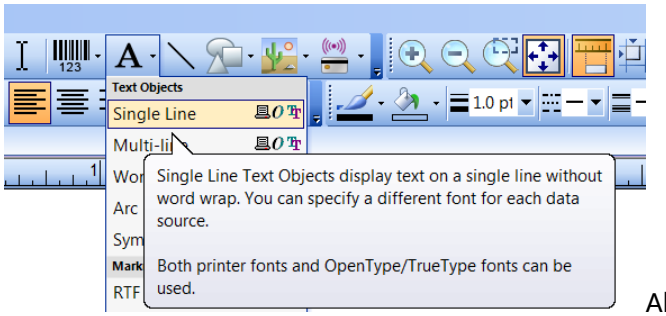
6- In Side Edges, Select “No, it does not” to answer the question if the stock has a small amount of unused material. Click Next.

7- In Printed Item Shape, Select Rectangle. Click Next.

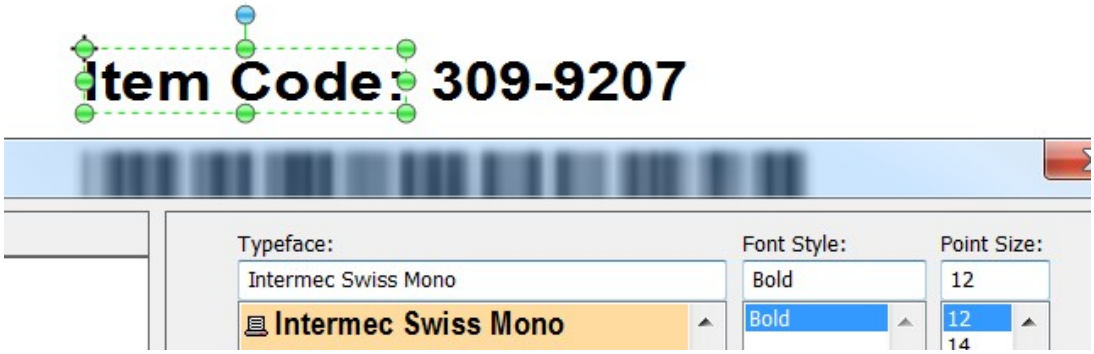
8- In template size enter the Width and Height of the label, in this case is W=6.5 in, H=4.0 in, select Landscape for orientation. For this example we Clicked Finish. The other screens does not need to be changed.



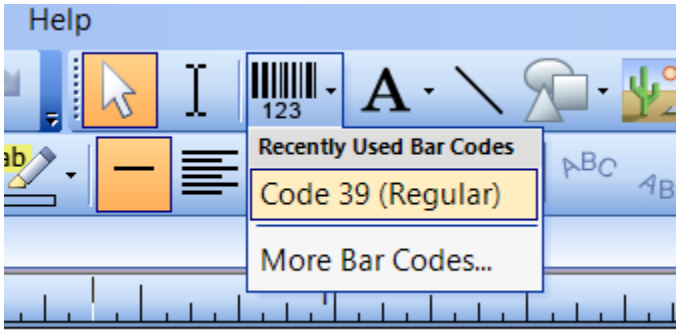
9- Create 2 texts and one barcode:

1. 

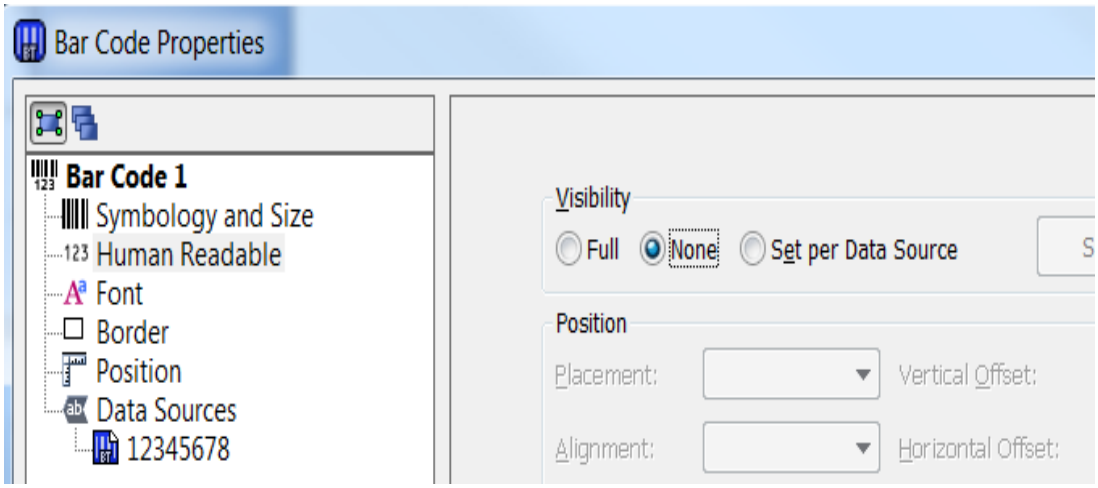
Always use printer internal fonts for text objects:



2. Create the bar code:



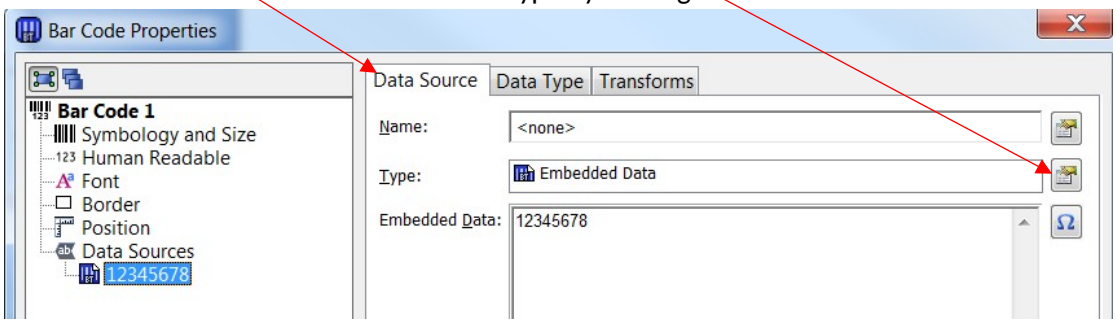
For the barcode, in properties, select Human Readable None:



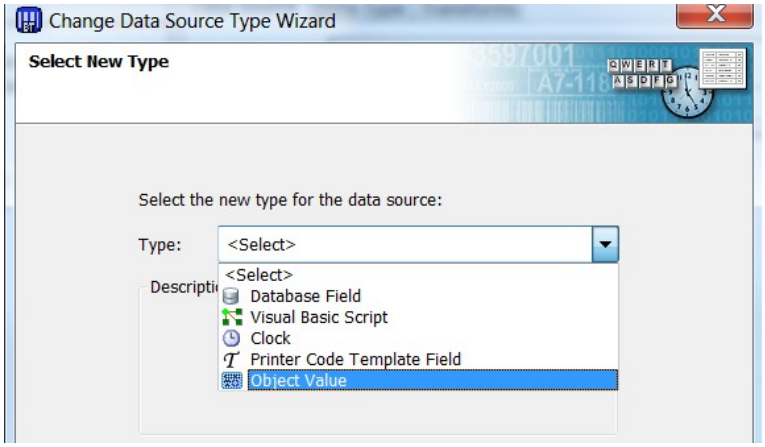
3. You should have your design like this:

b. Converting our static design to a dynamic design

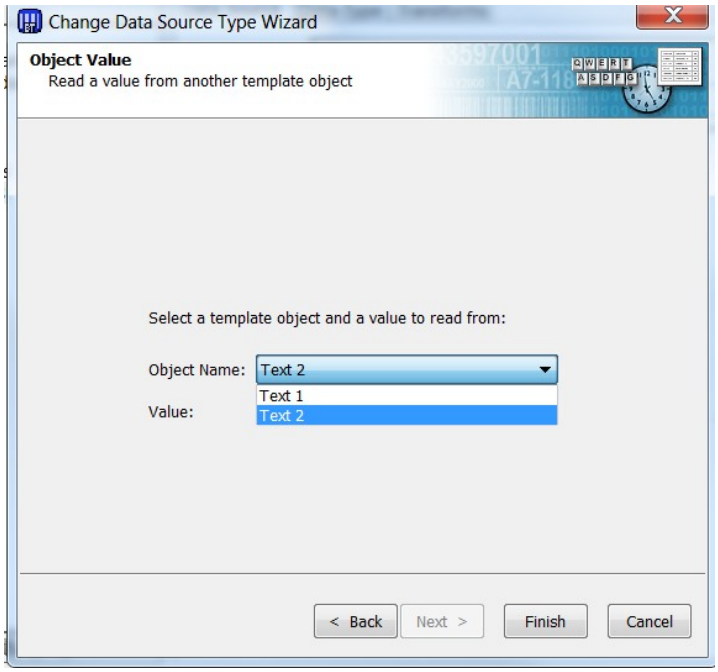
1- Double click on the barcode and under select Data Sources, select the one already there, select the Data Source tab and select new type by clicking here



2- In the next screen select Object Value and click Next.

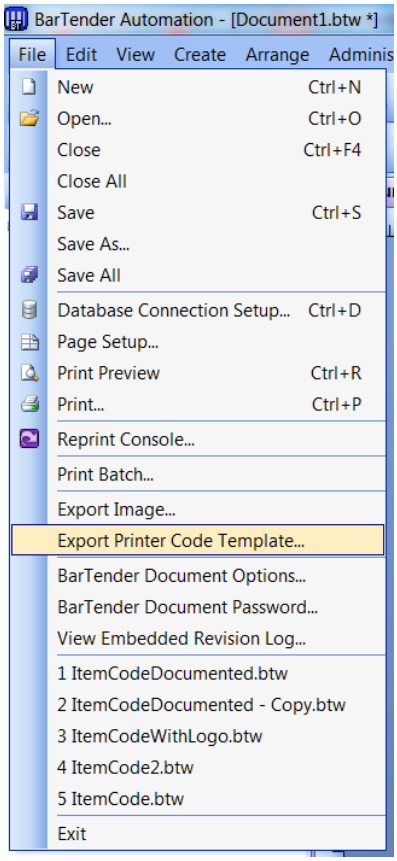


3- Assuming that “Text 2”, was created after the Item code label to the left. Click Finish and close the Barcode Properties screen

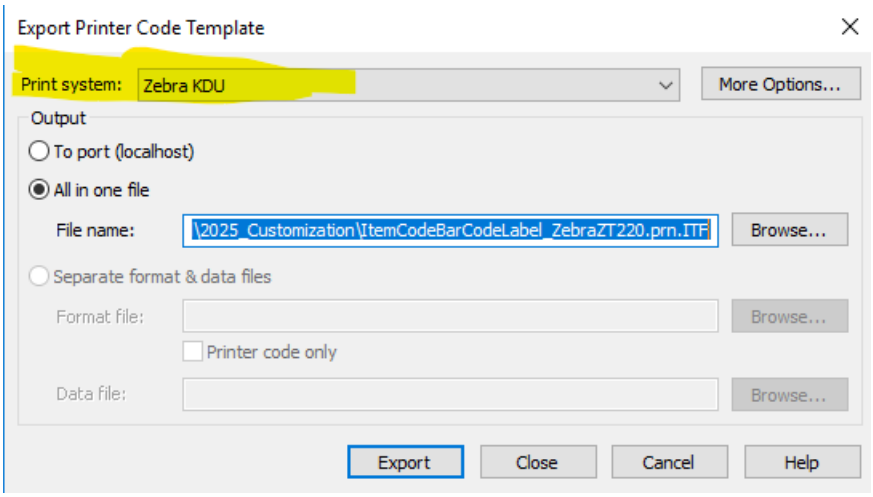


c. Exporting our dynamic label design.

1- In the Bartender menu, select File->Export Printer Code Template…



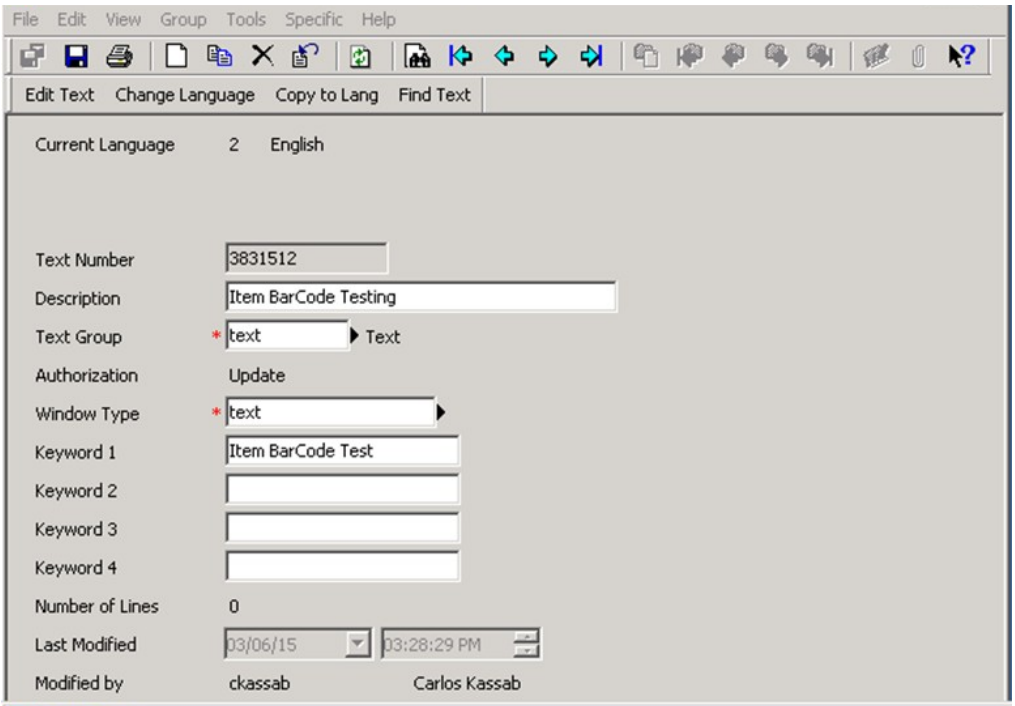
2- In the next screen, print system must be Zebra KDU, then browse the path and give a name to the template being exported and click Export.

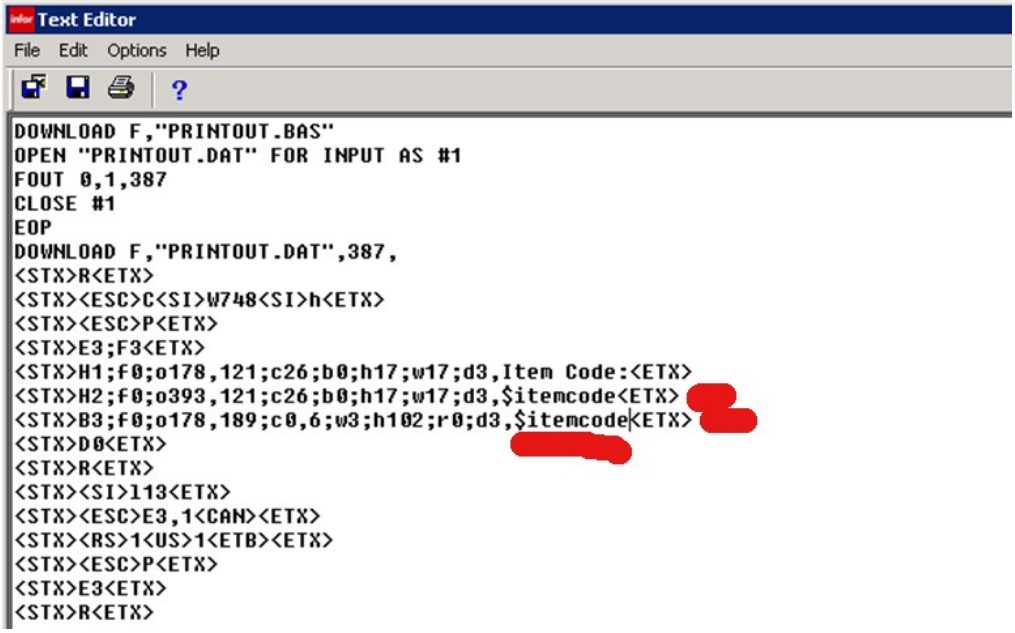


3- Using the exported template in LN to print barcodes for different items.
   A. If we open the template exported, it is a text file like the one below, we can see the item code. What we are going to do is to change this $ITEMCODE word replacing it with the item code provided by LN.

```
^XA
^SZ2^JMA
^MCY^PMN
^PW1201
~JSN
^JZY
^LH0,0^LRN
^XZ
^XA
^DFE:SSFMT000.ZPL^FS
^FT80,75
^CI27
^A0N,34,46^FDPN^FS
^FT138,117
^A0N,76,103^FD$ITEMCODE^FS
^FO98,147
^BY4,2.5^B3N,N,152,N,N^FD$ITEMCODE^FS
^FO64,308
^GB1040,0,2^FS
^FT98,353
^A0N,46,62^FD$ITEMDESC^FS
^FO64,366
^GB1040,0,2^FS
^FT98,409
^A0N,34,46^FDRev. Level^FS
^FT383,418
^A0N,51,69^FD$ITEMREV^FS
^XZ
^XA
^XFE:SSFMT000.ZPL^FS
^PQ1,0,1,Y
^XZ
```

B. The first thing to do is to create the text as text in LN copy the template and paste it in the LN text window, save and exit from the text editor, it is very important to write down the Text Number because we will use it to open the text and replace the label $itemcode, see the image below, session tttxt1100m000





d. Now we need to create a device in LN in order to be able to print labels. The actual sample is very simple but, what if we include a logo image in the printing?. According to my testing, the logo is exported as part of the template but, when I pasted the template in the LN text editor, the logo image code was completely changed and there was no logo printed so, I implemented a way to print labels including logo images.

    1- I created the template as I just did for this document. The template looks like this:

```
DOWNLOAD F,"ITEMCODE.BAS"
OPEN "ITEMCODE.DAT" FOR INPUT AS #1
FOUT 0,1,1439
CLOSE #1
EOP
DOWNLOAD F,"ITEMCODE.DAT",1439,
<STX>R<ETX>
<STX><ESC>C<SI>W1276<SI>h<ETX>
<STX><ESC>P<ETX>
<STX>E3;F3<ETX>
<STX>B1;f0;o165,288;c0,6;w3;h102;r0;d3,309-9207<ETX>
<STX>H2;f0;o152,195;c26;b0;h17;w20;d3,Item Code:<ETX>
<STX>H3;f0;o406,195;c26;b0;h17;w17;d3,309-9207<ETX>
<STX>D0<ETX>
<STX>R<ETX>
<STX><SI>l13<ETX>
<STX><ESC>E3,1<CAN><ETX>
<STX><ESC>g0<ETX>
<STX>!B}D® %É…Ê"%É„Ë"%É…Ê"%É„Ë"%É…Ê"%É„Ë"%É…Ê"%É„Ë"%É…Ê"%É„Ë"%É…Ê"%É„Ë"%É…Ê"%É„Ë"%É…Ê"%É„Ë"%É…
Ê"%É„Ë"%É…Ê"%É„Ë"%É…Ê"%É„Ë"%É…Ê"%É„Ë"%É…Ê"%É„Ë"%É…Ê"%É„Ë"%É…Ê"%É„Ë"%É…
Ê"%É„Ë"%Ä))))‡É"%Å))))„,)È"'¶Ûýÿÿÿÿÿÿ€€ ýÿÿÿÿÿ¿ÛÖŠ"'É¤òÿÿÿÿÿ¿‹€Àþÿÿÿÿÿß´●"<ETX>
<STX>&Ð©,$))ª))"&'ª›))ª„))),))),)"&¨¤©"&¨ ))§))"&¦¥¨"&¥«)¤ "&¥ª¨"& £¯¤"&£¬¦"& ¢²¤"&¡ ¯¦ "&¡³ ¢"&}))¡´¤"&Ž
))³))¢))"&})) ¶£"&Ž ·)) ))"&}))Ÿ·£"&Ž ¹¢"&}))ž¸))¢"&}¡¸£"&} º¢"&} º¢))"&} º¢"&} º¢))"&} ¼ "&} º¢))"&} º)) "&} »¡))"&}
º)) "<ETX>
<STX>&} º¢))"&} º¢"&} º¢))"&}¡¹¢"&}))ž¸¤"&Ž ¹¢"&}))Ÿ·))¡"&Ž ·¢))"&}))Ÿ·£"&Ž¡´))¢))"& ¡´£"&Ž¢³))¢))"&¡ ¯ ¤"&
¢²¤"& £¯¦"&¢¬£"&¥« ¦"&¤«¦"&¦§¨"&¦¢¨"&'¨¢))¦))"&'¨))œ))©))"&'ªžª"&'))«●))"&"«,)"¬))"&'))º)),)…))
±"&"¯))))ˆ),º"&"))³„³"&●´…±))"<ETX>
<STX>&–³„²"&–³…±))"&—²„º))"&™º…® "&˜))¯„º"&š¯…® "&œ„¯"&š))…¬"&›® „))"&ž«…ª"&¬„¬"&Ÿª…¨"&Ÿª„ª"&¡¨…
¦"&Ÿ))¨„¨"&£¦…¤"&¢))¥„¥))"&¦£…¡"&© „ "&§)) … "&«))œ„"&¬…›"&® ))™„™,)"&® ›…—))"&²))●„"))"&±))–
…"))"&¸))„ "&¶))'…))"&¿))))†„…))))ƒ)"&½,)))))………))),"<ETX>
<STX>(<ETX>
<STX><RS>1<US>1<ETB><ETX>
<STX><ESC>P<ETX>
<STX>E3<ETX>
<STX>R<ETX>
```

2- I created 2 files, one with just text and the other with the code for the image(in red above), the file with just text is like shown below.

```
DOWNLOAD F,"ITEMCODE.BAS"
OPEN "ITEMCODE.DAT" FOR INPUT AS #1
FOUT 0,1,1439
CLOSE #1
EOP
DOWNLOAD F,"ITEMCODE.DAT",1439,
<STX>R<ETX>
<STX><ESC>C<SI>W1276<SI>h<ETX>
<STX><ESC>P<ETX>
<STX>E3;F3<ETX>
<STX>B1;f0;o165,288;c0,6;w3;h102;r0;d3,$itemcode<ETX>
<STX>H2;f0;o152,195;c26;b0;h17;w20;d3,Item Code:<ETX>
<STX>H3;f0;o406,195;c26;b0;h17;w17;d3,$itemcode<ETX>
<STX>D0<ETX>
<STX>R<ETX>
<STX><SI>I13<ETX>
<STX><ESC>E3,1<CAN><ETX>
<STX><ESC>g0<ETX>
***CompanyLogo***
<STX>(<ETX>
<STX><RS>1<US>1<ETB><ETX>
<STX><ESC>P<ETX>
<STX>E3<ETX>
<STX>R<ETX>
```

3- I created in LN the text showed in the previous step in order for the item code to be replaced by the report script and the company logo will be replaced by the code for the image created in another file, for this replacement I am using the next console C# program:

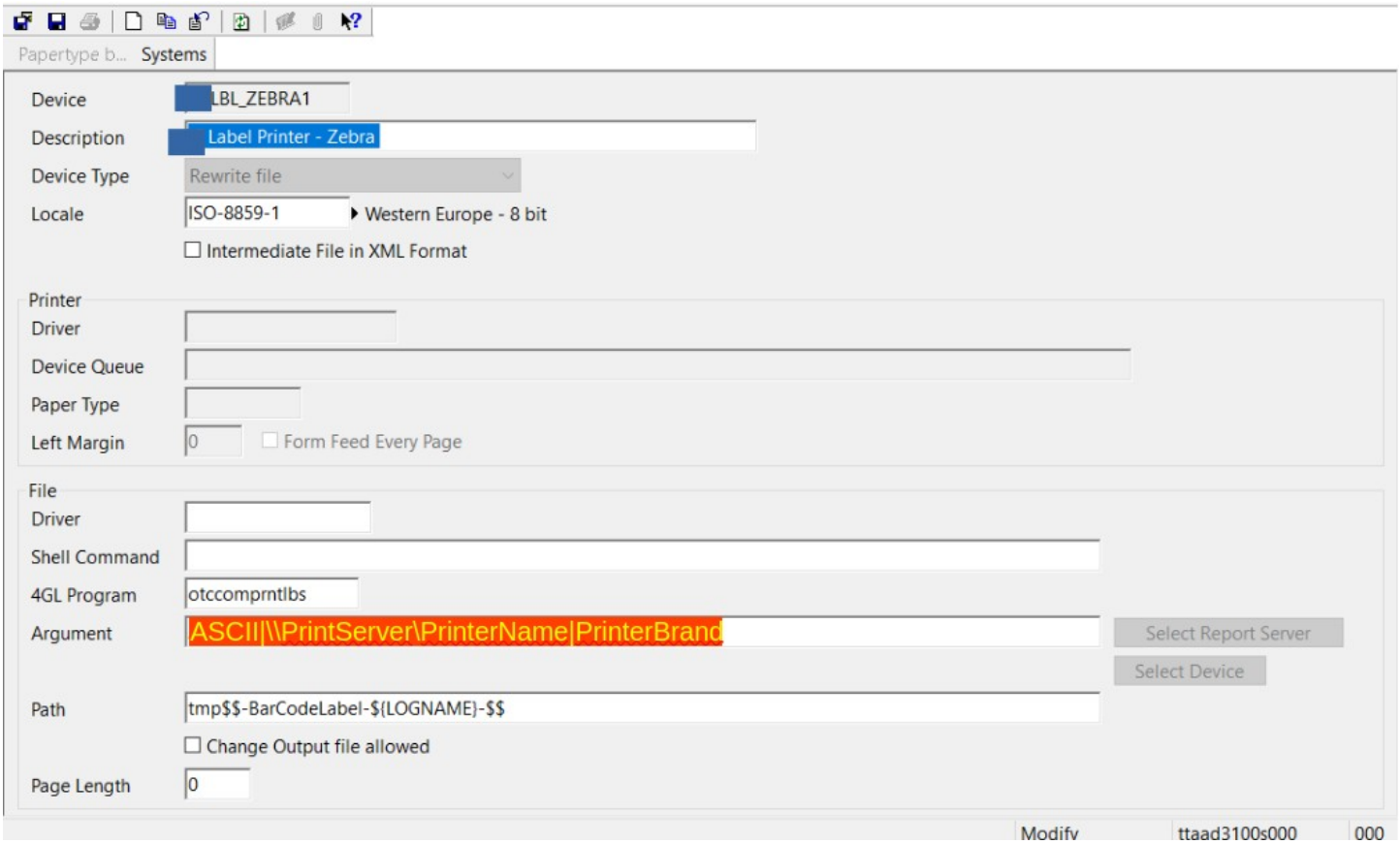
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.IO;

namespace PrintBarCodeLabelWithLogo
{
   class Program
   {
      static void Main(string[] args)
      {
         String LogofileName = "";
         String SpoolFile = "";

         if (args == null)
         {

         }
         else
         {
            LogofileName = args[0];
            SpoolFile = args[1];
            string SpoolToPrint = args[2];

            string LogoFile = File.ReadAllText(LogofileName, ASCIIEncoding.Default);

            string SpoolFileToPrint = File.ReadAllText(SpoolFile, ASCIIEncoding.Default);
            string DataWithLogo = SpoolFileToPrint.Replace("***CompanyLogo ***", LogoFile);
            File.WriteAllText(SpoolFile, DataWithLogo, ASCIIEncoding.Default);
            File.Copy(SpoolFile, SpoolToPrint);
         }
      }
   }
}
```

4- Creating the device in LN

1. First I created a 3GL program:

```
|*****************************************************************************
|* Title : tccomprintlb – Device to print labels by using ttstpconv and external program
|* Author        : Carlos Kassab
|* Date          :
|*****************************************************************************
|* Script Type   : 3GL
|****************************** DECLARATION SECTION ***************************

        extern  domain tcmcs.str132    spool_file_name, program_name
        extern  domain tcmcs.str132    prog_arguments
        extern  domain  tcmcs.long     retval
        extern  domain  tcmcs.long     fileid1
        extern  domain  tcmcs.long     err
        extern  string                 file_type(33), print_queue(100)
        extern  string                 printer_brand(100)

        #pragma used dll ottdllbw
        |#include <bic_tt>

function main()
{
        | message(argv$(3)) | It contains the file type and the printer queue
        string.scan(argv$(3), "%s|%s|%s", file_type, print_queue, printer_brand )

        spool_file_name = creat.tmp.file$( bse.tmp.dir$() )
        program_name = "E:\Labels\PrintBarCodeLabelWithLogo.exe"
        prog_arguments = "E:\Labels\CompanyLogo.prn " & spool_file_name
        prog_arguments = prog_arguments & " " & print_queue

        | Convert the report file into the temp file.
        retval = wait.and.activate("ttstpconv", argv$(1), spool_file_name, file_type, argv$(4))
                | ttstpconv is a conversion program in BaaN/LN tools.
                | Arguments are passed from BaaN/LN:
                |    1 = Name of input (temporary) file contain dev independent report
                |    2 = Name of output file ===>>> spool_file_name
                |    3 = Additional arguments (from Device data) ===>>> file_type
                |    4 = Specilar parm for ttstpconv (unknown reason)

        |Set the company logo and send file to the printer
        retval = run.prog(program_name, prog_arguments, RP_WAIT)
        retval = seq.unlink(spool_file_name)
}
```

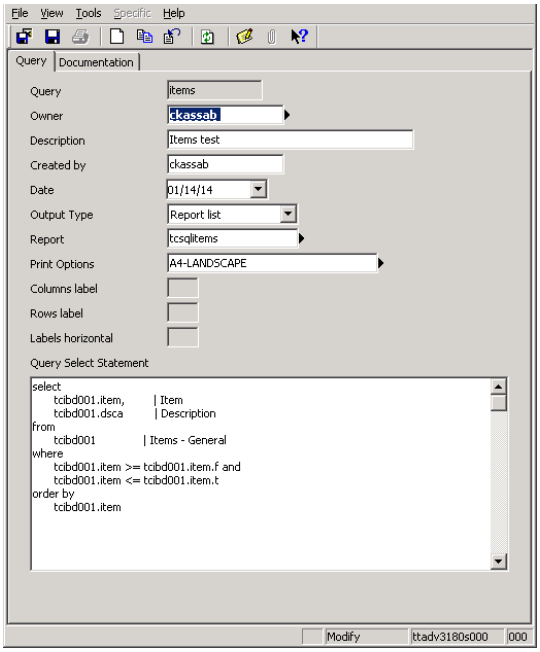
2. Now I created the LN device(device data session ttaad3100s000):



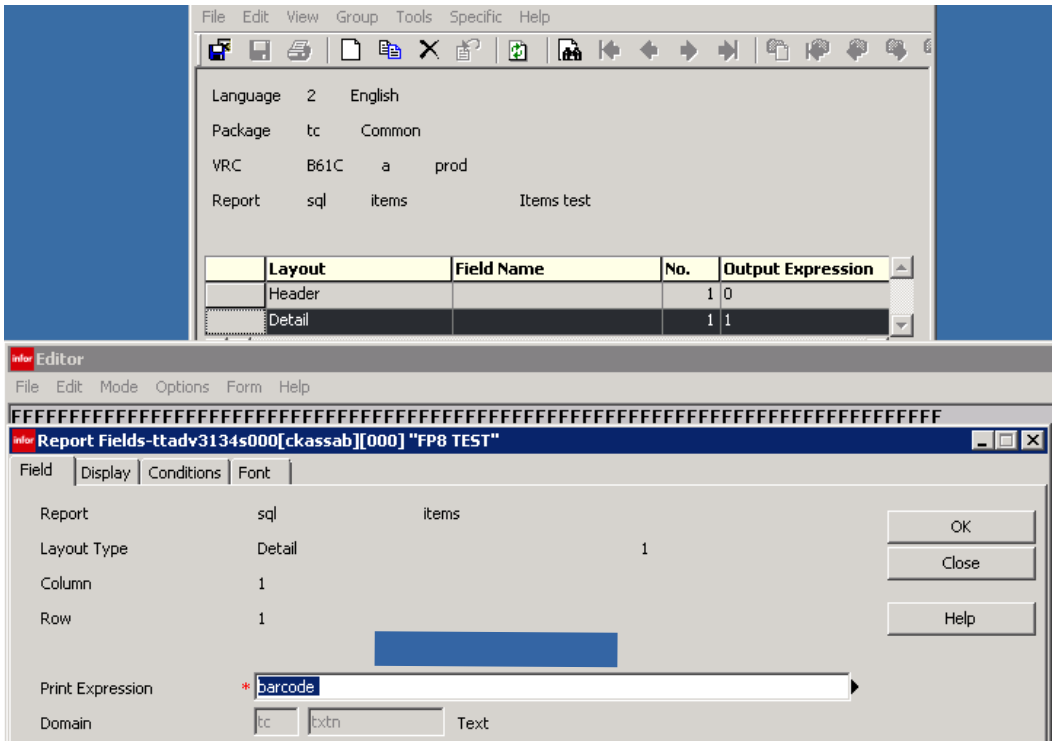
It is important to see the argument field above, it has 3 parts, one is the file format for the program ttstpconv, the printing queue were the file should be printed, and the printer brand(Sometimes used to decide actions based on this).

e. Now that we have our template and our LN device, we just need to create a report in LN to print labels.

1- To keep it simple we are going to create a query in LN(query data:ttadv3180s000):



2- Edit the report first by avoiding printing the header and just with one detail line, in the detail line just one text pointer variable, here the report Layouts(session ttadv3531m000):



3- The report script for this report should look like this:

```
|*****************************************************************************
|* tcsqlitems   VRC B61C a  prod
|* Items test
|* ckassab
|*****************************************************************************
|***************************** declaration section ***************************
declaration:

        extern  domain tcitem  itemcode |-> Variable should have the same name as in the LN text $itemcode
        extern  domain tctxtn   barcode   |-> This is our text we are going to print

before.program:
        lattr.textexpand=TRUE   |->This will tell LN to replace the variables in the text for the values here.

detail.1:
before.layout:

        barcode = 4889835| Testing text code for barcodes. My text containing the logo had this number.
        itemcode = strip$(shiftl$(tcibd001.item))
```

IMPORTANT NOTE. By assigning the value to itemcode variable, LN will replace $itemcode in the text pointed by the variable barcode with the itemcode actual value.

To print, we just run our query and select our new device.

That is all we need to print labels from LN. We can use this printing method with any printer brands like Zebra.