



WYSIWYG Spine Labels

A User Friendly Printing Alternative

Andy Helck

Wilkinson Public Library, Telluride Colorado

Introduction

- In 2011 our library undertook a major re-cataloging our children's materials. We did away with Dewey decimal numbers in our juvenile non-fiction books and our picture books. Our staff created categories and subcategories for a new set of call numbers for the collection.

**JP
BASICS
COLORS
TEC**

**JP
ANIMALS
BIRDS
BRU**

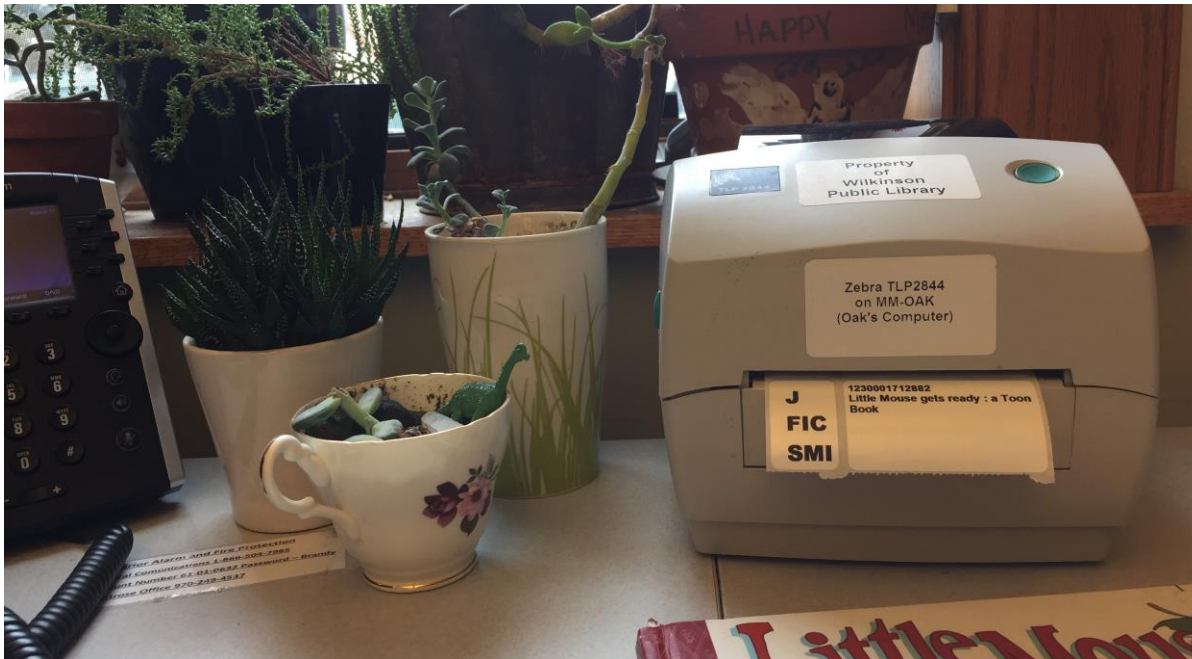
At this time, with several thousand books to relabel, the shortcomings of the Millennium system became apparent!

Our Hardware choice

We looked around to see what printers other libraries were using. We wanted a waterproof label and we were not afraid to pay a premium for the labels.

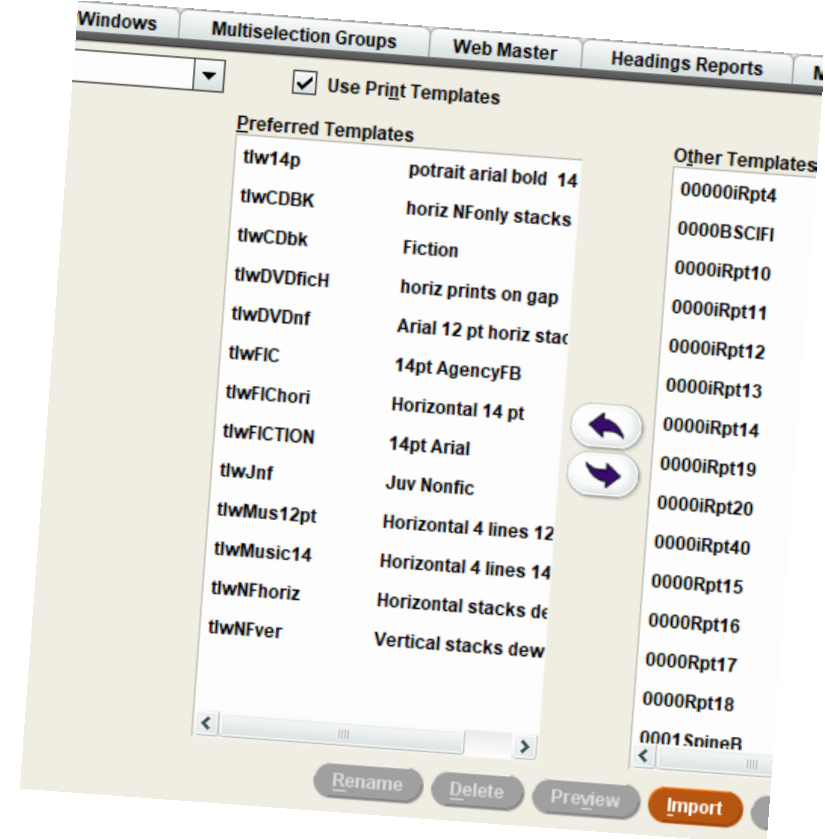
We bought three Zebra brand TLP2844 desktop printers. These machines are running strong nearly 10 years later.

If we were buying again today we might look at the Zebra ZD620



Native Printing in Millennium/Sierra

- The typical setup is to create Print Templates with a variety of font styles, font sizes, page orientation, etc.
- The cataloger opens the record and selects from the list of templates. There is no preview, so its not uncommon to have to try several different templates.
- Templates can be edited, and margins and font sizes adjusted. But this has to happen outside of Millennium or Sierra, and the modified template imported and installed.
- Many clicks can be required to create a single usable spine label.



Many Choices, Limited Satisfaction

Anatomy of a Print Template

- Right Click on the exported template file and “Open with Notepad”
- You’re looking at an XML file which defines many, many more text areas and headers and banners than you need for a simple spine label. You can delete empty document elements
- The beginning of the file defines multiple fields. These are the data from Sierra that you can apply to your document.
- There is only a single text element of interest, a <detail> toward the very bottom

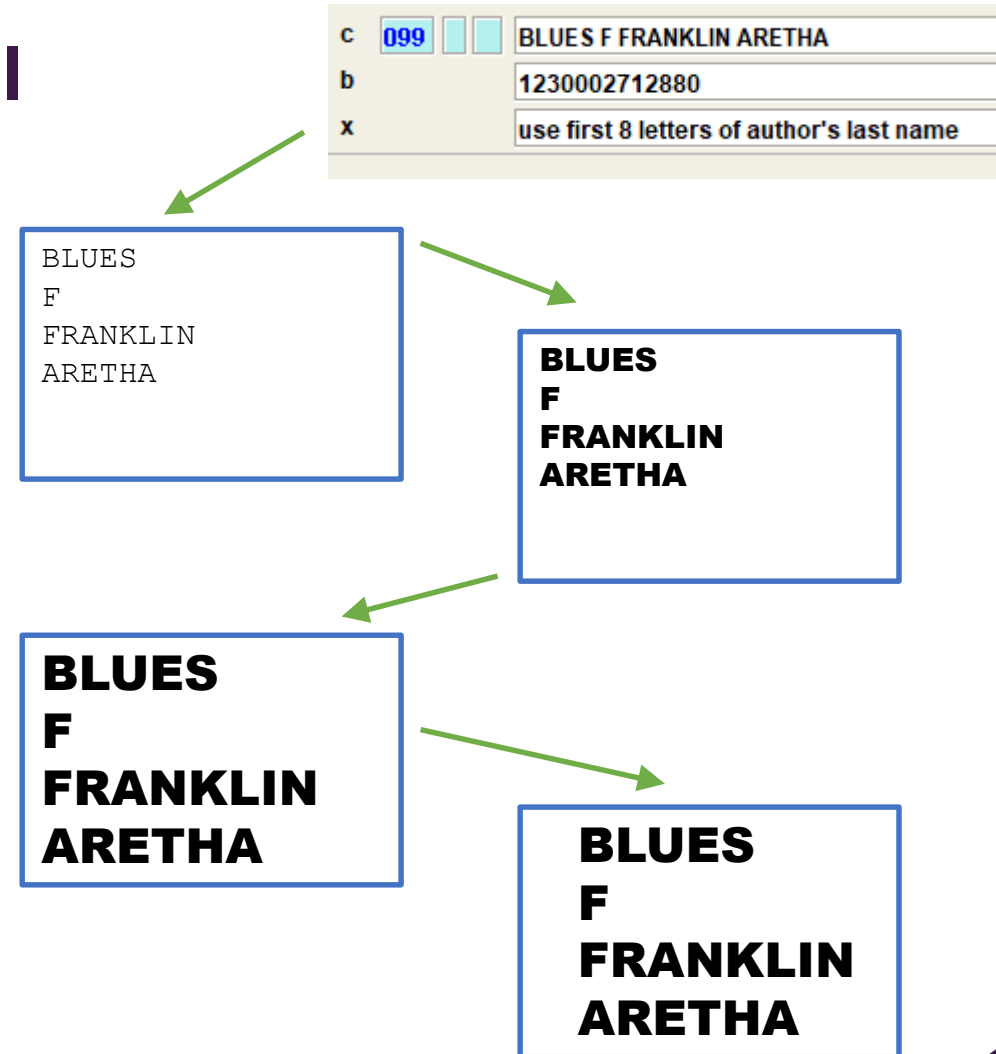
```
</field> <field name="callNumericStart"
class="java.lang.String">
  <fieldDescription><![CDATA[Data Element 2:
subfield a/h (1st occurrence only of subfield a), numerical
content until decimal point, from c-tagged varfld Purpose:
call number]]></fieldDescription> </field>
```

```
</field>
<detail>
  <band height="135" splitType="Stretch">
    <textField isBlankWhenNull="false">
      <reportElement key="textField-1" x="24" y="15" width="80" height="100">
        <textElement verticalAlignment="Bottom" rotation="Right">
          <font fontName="Arial" size="12" isBold="true" pdfFont="Arial-BoldMT">
            </textElement>
            <textFieldExpression class="java.lang.String"><![CDATA["CDBK"]></textFieldExpression>
          </textElement>
        </textElement>
      </reportElement>
    </textField>
  </band>
</detail>
```



What it takes to print a spine label

- Identify and break up the parts of the call number...*parsing*
- Decide how to arrange those parts on separate lines...*stacking*
- Pick a font, size and style...*formatting*
- Allow the user to maximize the font size as needed...*filling*
- Center the text in the printed label as accurately as possible...*margins*



Nothing ventured, Nothing gained

I'd been programming since my school days and in my previous career, mostly in obscure languages like 6809 Assembler and Ladder Logic for motion controllers. I wanted to learn a modern language object-oriented and I had used Turbo C back in the 1990s.

C++ was too scary, so I decided to learn the Microsoft language called "*C Sharp*."

A Windows Form Application is the classic Windows program that offers a graphic user interface. This means buttons, textboxes, menus, etc. These controls include the *RichTextBox* which is like a miniature version of Microsoft Word.

Inside a *RichTextBox* our text can be arranged into paragraphs, indented, italicized and formatted. It has turned out to be a great choice as a spine label editing tool!

A Tale of Two Situations

In a way, I traded difficulties. In Millennium and Sierra, the data we needed – call numbers – was ready at hand. But the fine level control over formatting, rotation and font was missing. My program gave great control over the finished label, but getting the call numbers into the program became the hurdle.

I started out with the simplest tools. Either just type the call number in correctly by hand, or copy and paste from the item record into the program. Simple rules to parse and stack and remove the subfield indicators were implemented.

I then added the ability to import a text file of barcodes, titles and call numbers. These files were of course generated in the *Create Lists* module. The user could now search by barcode and bring up the call number and title for printing.

Most recently I've added the ability to access the database directly over the internet using *SierraAPI* technology. It takes a few seconds to retrieve each record, but the convenience is worth it!

Library Labels

Open File Select Printer

Preview and Print Printer Setup File Import Settings SierraAPI Help Pages Event Log

**DVD
COMEDY
REV**

Portrait

[Arial 23 Bold](#)

Font Size

<A <A

[Go To Font Style Edit](#)

☒ Auto Margins

<<Lower <<Indent <<Outdent <<Raise

1230001252628
Revenge of the Pink Panther

[Arial 12 Bold](#)

Print Zebra TLP2844

Search

Next

CALL #(ITEM)	BARCODE	TITLE
DVD DRAMA C CABARET	1230001251004	Cabaret [videorecording] / an ABC Pictures Corp...
DVD DRAMA PER	1230001252593	Perfume [videorecording] : the story of a murdere...
DVD COMEDY REV	1230001252628	Revenge of the Pink Panther [videorecording] / ...
DVD DRAMA ROM	1230001252655	Romance & cigarettes [videorecording] / United ...
DVD DRAMA CIN	1230001252691	The Cincinnati Kid [videorecording] / Metro-Gold...
DVD CULTURE UNITED STATES DEVIL'S	1230001252726	Devil's playground [videorecording] / Wellspring i...
DVD SCIENCE TECHNOLOGY TRA	1230001252799	Transforming energy [videorecording] : [what is o...
DVD DRAMA HEN	1230001251102	Henry & June [videorecording] / Universal Pictur...
DVD POLITICS & GOVERNMENT PRESIDENT...	1230001251139	Barack Obama [videorecording] / produced by T...
DVD ECONOMICS & BUSINESS BEN	1230001251246	Ben & Jerry's [videorecording] / produced by To...
DVD POLITICS & GOVERNMENT COCALERO	1230001250942	Cocalero [videorecording] / Fall Line Films ... [an...
DVD BIOGRAPHY CHE	1230001251077	Che Guevara where you'd never imagine him [vi...
DVD HISTORY WORLD IRA	1230001251148	Iran [videorecording] : the next Iraq?.
DVD BEST PICTURE L LAST	1230002434654	Bernardo Bertolucci's The last emperor [videorec...
DVD DRAMA MAT	1230001249651	The matrix revolutions [videorecording (DVD)] / ...

We've just searched for barcode **1230001252628** and because I have a large file of all our DVD items loaded, we found the movie we were looking for.

Before I print the label I will enlarge the text and perhaps try some different Arial fonts.

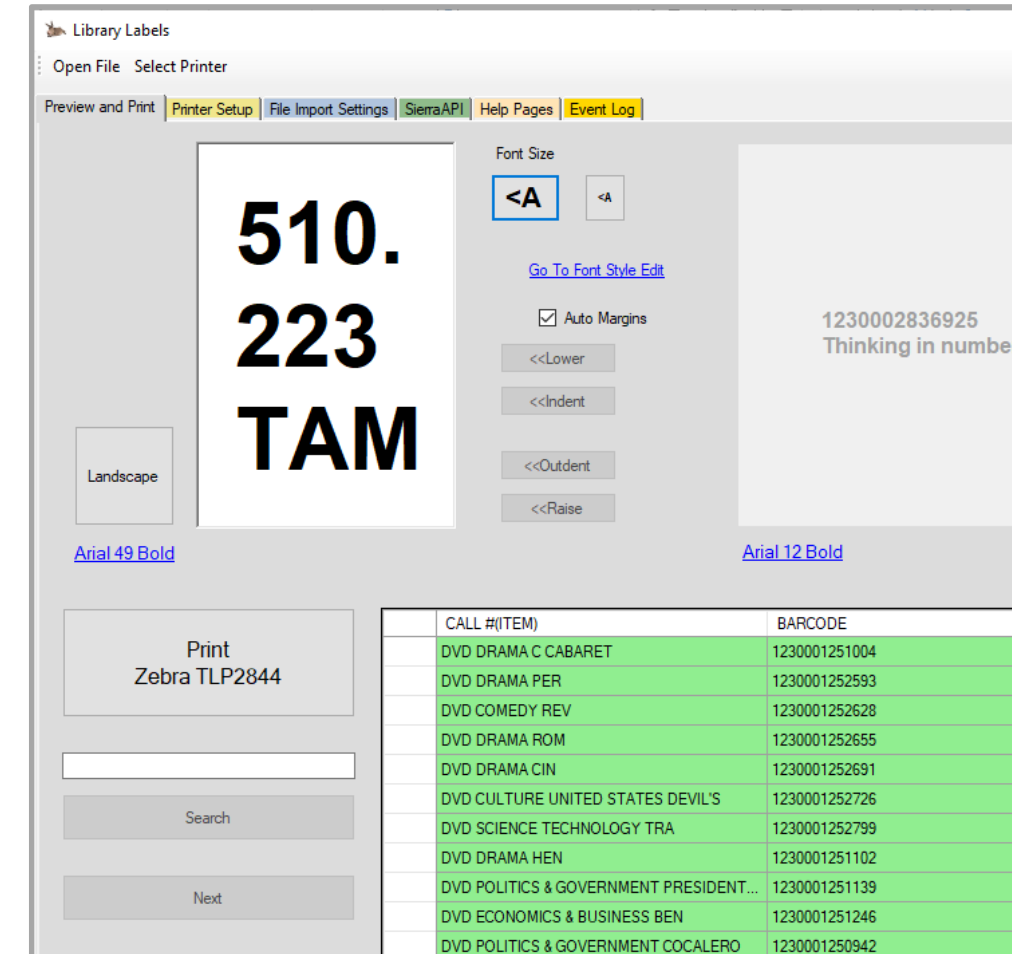
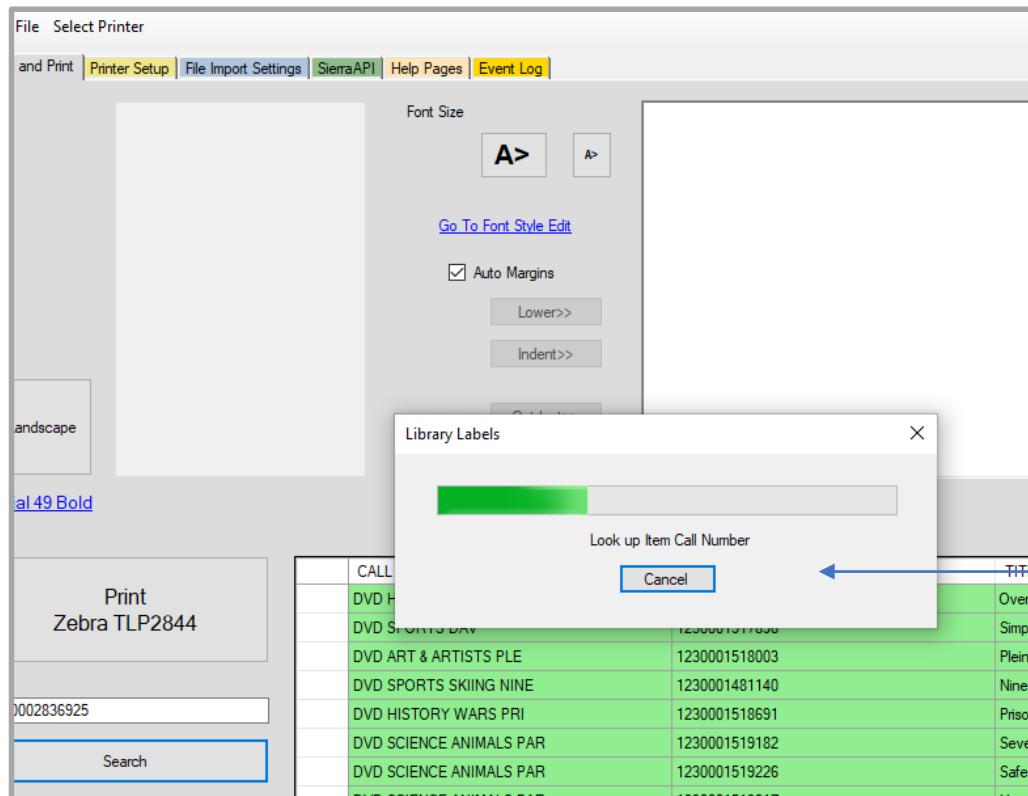
Our library doesn't use the pocket label on the right, but it too can be formatted and sized.



#IUG2020



This next item is non-fiction book, and its spine label fits better in Portrait orientation. The Dewey number has been stacked into 3 lines according to some basic parsing rules.



As this was not a DVD, the program had to look for the information using the new API technology from III. It only takes a few seconds to load a record and get the call number!

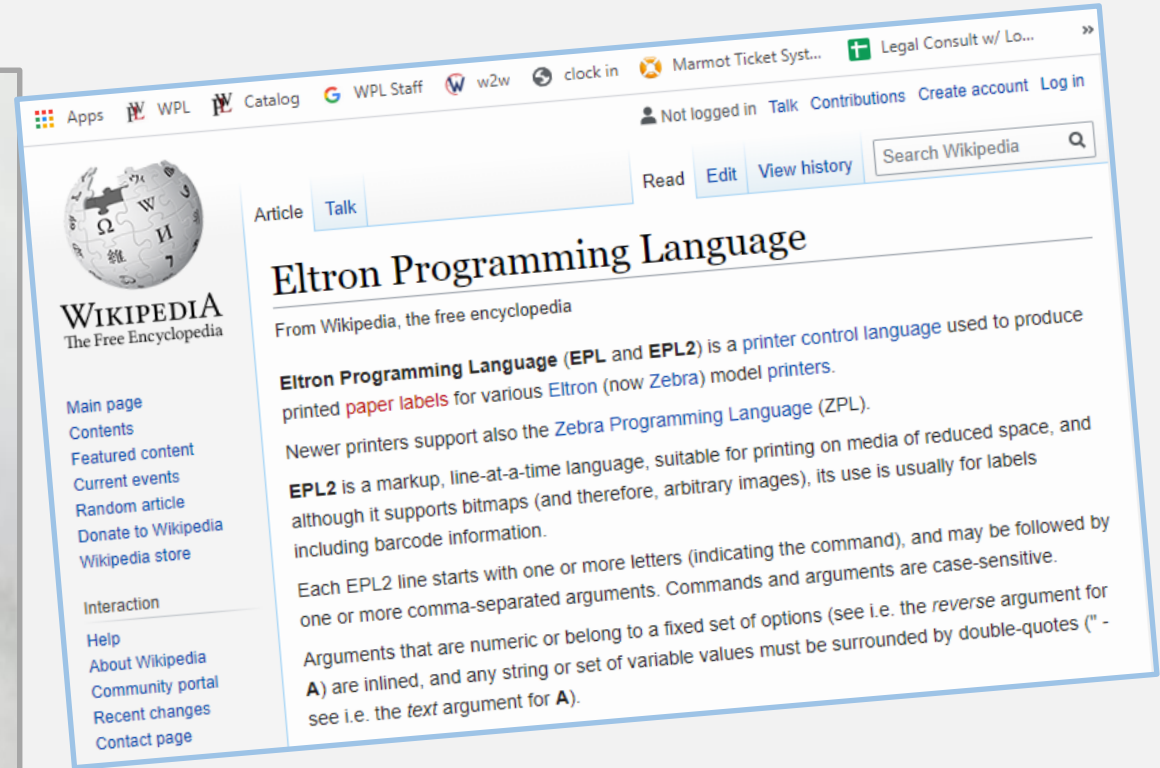
Part 2

How the Code Works!

Thank Goodness for Old School Computing!

The Zebra printers are known in the industry as barcode printers. Companies that ship a lot of packages use these to print UPS and Fedex labels by the thousands, and they have been in use for decades.

The original software interface goes back years, to something called **Eltron Programming Language**, or **EPL** and **EPL2**. Newer and more complicated languages, like **Zebra Programming Language ZPL** exist, but the older **EPL** is supported in all the newest Zebra printers.



#IUG2020



Simple Graphics for Simple Programming!

Google will instantly return results for “EPL2 Programming Manual” I found the **GW** command after an hour’s reading

GW Command - Direct Graphic Write

Description Use this command to load binary graphic data directly into the *Image Buffer* memory for immediate printing. The printer does not store graphic data sent directly to the image buffer.

The graphic data is lost when the image has finished printing, power is removed or the printer is reset. Commands that size (**Q** and **q**) or clear (**N** and **M**) the image buffer will also remove graphic image data.

Syntax **GW***p*₁.*p*₂.*p*₃.*p*₄**DATA**↵

Parameters **p**₁ = Horizontal start position (X) in dots.

p₂ = Vertical start position (Y) in dots.

p₃ = Width of graphic in bytes. Eight (8) dots
= One (1) byte of data

p₄ = Length of graphic in dots (or print lines).

DATA = Raw binary data without graphic file formatting. Data must be in bytes. Multiply the width in bytes (**p**₃) by the number of print lines (**p**₄) for the total amount of graphic data. The printer automatically calculates the exact size of the data block based upon this formula.

WYSIWYG means “What You See Is What You Get.” At one time this was considered something of a radical idea, now of course we are all so used to computers instantly and perfectly showing us the “print preview” or the finished web page as we are editing the HTML that we take it for granted!

In order to provide this feature in my program, I sized the textboxes in which you compose the labels to be the exact same size (in pixels) as the paper label that is being printed.

The Zebra prints 203 dots per inch (dpi) whereas the traditional Windows display is the much coarser 96 dpi. The textbox looks big on the screen, which makes it easier for my aging eyesight!



#IUG2020



Common Ground: the Windows Bitmap

A simple color bitmap underlies all of windows graphics programming, and its exposed in C# in a very straightforward manner.

Essentially what happens when we hit *PRINT* is that the program captures the bitmap from the textbox and sends it to the printer using the GW command.

The one twist is if we print in Landscape mode, the program has to rotate the color bitmap. That's because the textbox on the screen keeps its text horizontal (but changes dimension) where in reality the printer will apply the text vertically. I easily found code examples online to do this.

This basic bit of coding went quite quickly. I soon had a basic program with a single textbox that would print in various fonts and both orientations.

Well I'd be fibbing if I didn't admit there were some difficulties along the way. The Windows bitmap is 4 bytes per pixel for full color, the printer only needs 1 byte per pixel.

More vexing is the fact that the RichTextBox is not a native Windows control (its part of .NET winforms) so I had to look online for a ToBitmap() method. But all the code is written, and its available on GitHub to download and do with as you like!

C# is an object oriented language and it's a bit more rigid than JavaScript, so there is a learning curve having to do with classes and objects.



Dealing with Create Lists...

By the time I wrote this program, I had been dealing with the vagaries of the Create Lists *Export* function in Millennium and Sierra. If anyone has wrestled with getting Excel or other programs to correctly parse the various delimiters, text qualifiers, repeat field indicators, and prevent collisions with obscure MARC21 characters, they will know what I am talking about.

So when I added the ability to import a large file of CSV data from Millennium, I decided to embed a utility that would not get fooled by weird syntax!

In the Import Tab you can directly edit the text file if you have to, or alter the parsing algorithm, and immediately observe the results in the middle panel, called the data grid!

The screenshot shows the 'Library Labels' application window. The 'Data Format' tab is active, displaying options for parsing a CSV file. The 'Specify First Column Name' radio button is selected, with the 'First Column Header' set to 'CALL #(ITEM)'. The 'Default RFD' is set to '1'. A 'ReLoad Raw Data' button is visible. Below the settings is a data grid with the following columns: CALL #(ITEM), 245_a, BARCODE, and ORDER CODE. The grid contains 10 rows of data, including items like 'JP SONGS CRE', 'J FIC CAP', 'J FIC ZIE', 'J FIC NEA', 'J FIC TAY', 'J FIC SNO', 'J FIC BRI', 'J FEELINGS FRO', and 'J TRANSPORTATION GROUND LIN'. Below the data grid is a text area showing the raw CSV data being imported, with the first line highlighted in blue: 'JP SONGS CRE|the neighborhood singalong / a treatise on a capella singing|1230002896771|FTP0GNX2'.

CALL #(ITEM)	245_a	BARCODE	ORDER CODE
JP SONGS CRE	the neighborhood singalong / a treatise on a ca...	1230002896771	FTP0GNX2
J FIC CAP	panda kisses	1230002896833	FTP0GNX2
J FIC ZIE	sleepy dog wake up!	1230002896888	FTP0GNX2
J FIC NEA	so many mes	1230002896904	FTP0GNX2
J FIC TAY	water everywhere!	1230002896913	FTP0GNX2
J FIC SNO	eat your peas louise!	1230002896922	FTP0GNX2
J FIC BRI	clifford makes the team	1230002896931	FTP0GNX2
J FEELINGS FRO	feeling happy	1230002896851	FTP0GNX2
J TRANSPORTATION GROUND LIN	fire trucks	1230002896806	FTP0GNX2

```
JP SONGS CRE|the neighborhood singalong / a treatise on a capella singing|1230002896771|FTP0GNX2
J FIC CAP|panda kisses|1230002896833|FTP0GNX2
J FIC ZIE|sleepy dog wake up!|1230002896888|FTP0GNX2
J FIC NEA|so many mes|1230002896904|FTP0GNX2
J FIC TAY|water everywhere!|1230002896913|FTP0GNX2
J FIC SNO|eat your peas louise!|1230002896922|FTP0GNX2
J FIC BRI|clifford makes the team|1230002896931|FTP0GNX2
J FEELINGS FRO|feeling happy|1230002896851|FTP0GNX2
J TRANSPORTATION GROUND LIN|fire trucks|1230002896806|FTP0GNX2
J FIC MEI|tiny goes to the library|1230002896860|FTP0GNX2
J FIC ADE|five teddy bears|1230002896799|FTP0GNX2
J FIC GRA|bad luck lucy!|1230002896780|FTP0GNX2
JP STORIES FAIRYTALES CRE|jack and the beanstalk|1230002896762|FTP0GNX2
J FIC MEI|tiny goes camping|1230002896842|FTP0GNX2
```



Part 3

More Features that make Users Happy

Preview and Print | **Printer Setup** | File Import Settings | SierraAPI | Help Pages | Event Log

(13, 10)

^
move up

<< move left move right >>

move down
v

(228, 10)

^
move up

<< move left move right >>

move down
v

Zebra TLP2844

Perform AutoSense Procedure

Factory Defaults

Set Page Size

Check Alignment

Print Test Pattern

View Last Print Job

Spine Label Options

☒ Remove SubField Indicators

☒ Stack Spaces

☒ Stack Dots

☒ Unstack Ampersands

☐ Word Wrap

Help

Default Fonts

[Arial 18 Regular](#) Regular

[Arial Narrow 18 Bold](#) Narrow

[Arial 18 Bold](#) Bold

[Arial Black 18 Bold](#) Black

Help

Pocket Label Options

☒ Fill Pocket Data

☒ Sierra Field Cleanup

☒ Expand Escape Chars

☒ First Repeated Field Only

☐ Include All Repeated Fields

☒ Word Wrap

Help

The Zebra printer has a fairly loose guide system when it comes to holding the roll of labels left to right. On the other hand the machine is very precise about detecting the top and bottom edges of the label each time it advances the paper.

When printing from Millennium, we used to hold our thumbs against the edge of the labels to bias the print job one way or another.

The above printer settings tab allows the user to fine adjust the actual position of the print image on the label. Some adjustment is required, especially after replacing the roll or changing the ribbon.

Printer Alignment and Stacking



#IUG2020



Preview and Print | Printer Setup | File Import Settings | SierraAPI | Help Pages | Event Log

Sierra API Settings

Server Address

API Key

API Secret ☐ Show Secret

Barcode Regular Expression

This is the newest feature I've added and in some ways the coolest.

API is a general industry term for “connect to the database without using SQL queries” and Innovative Interfaces has been working very hard to make these effective and easy to implement.

You will need to enter credentials in this window that will authenticate to your library system. You will need an “API Key and Secret” that your admin can provide you with. Make sure you get permissions on Item and Bib records.

Access is read-only so there is no way to accidentally change your records!

Talk to your System Administrator...





*Pick your Fonts!
Pick your Size!
Pick your Orientation!
Look before you Print!
No more surprises!*

*The rectangles in the first few labels
are test patterns that help you align
the image with the paper labels*

Labels, labels and more labels...

Thanks for Coming!

