



**BellHawk v4.7 Users Manual**  
**Using the TAG Label Printing Module**

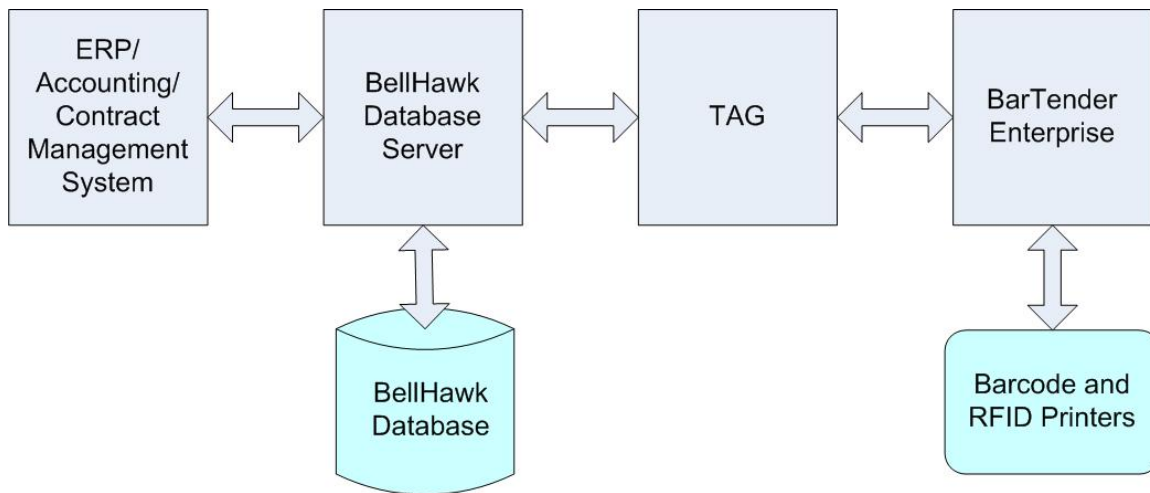
**Table of Contents**

Table of Contents.....	1
Introduction.....	2
Operator Interface.....	3
Printing Shipping Labels.....	3
Printing Product Labels.....	6
Printing Product Labels with Serial Numbers.....	10
Printing Product Container Barcodes.....	11
Printing General Container Barcodes.....	12
Setting-Up Label Formats.....	13
Introduction.....	13
Label Names.....	13
Field Name Association.....	15
System Attributes.....	18
Control Attributes.....	19
iCodes.....	19
Setting Up Returned iCodes in BarTender.....	20
System Attribute Values Available.....	27
For Product , Product Box and General Container Labels.....	27
From Container Type Records.....	27
From Item Master Records.....	27
From DBA Object.....	28
From Host DBA Object.....	29
Attributes for Commercial Shipping Labels.....	30
Setting Up Military Shipping Labels.....	32
The Debug Capability.....	35
Supporting Materials.....	37
Issues with BarTender.....	37
Labels Not Printing from BellHawk.....	37

## Introduction

The BH-TAG option in BellHawk enables users to produce:

- Product labels, including those with 2D Datamatrix and other specialized barcodes for serialized products, including Department of Defense (DoD) tracking. These labels can contain UPC/EPC/GTIN barcodes as well as manufacturer's name and address, part number and description, lot and serial number information, weight, size, and other characteristics. They can also contain Hazmat, RoHs and similar regulatory information.
- Product Box Labels, which can contain the same information as for the products but may also include count of products in the box. These labels may also include embedded RFID tags. This includes DoD National Stock Numbered (NSN) boxes with RFIS tags.
- General Container labels with tracking barcodes.
- Shipping Labels including commercial UCC 128 labels with GS1 SSCC tracking barcodes and DoD Military Shipping Labels (MSLs) with TCN and PDF417 (2D) barcodes. These labels can include contract information, ship to and from information as well as information about the pallet or other container to which the shipping barcode is attached.



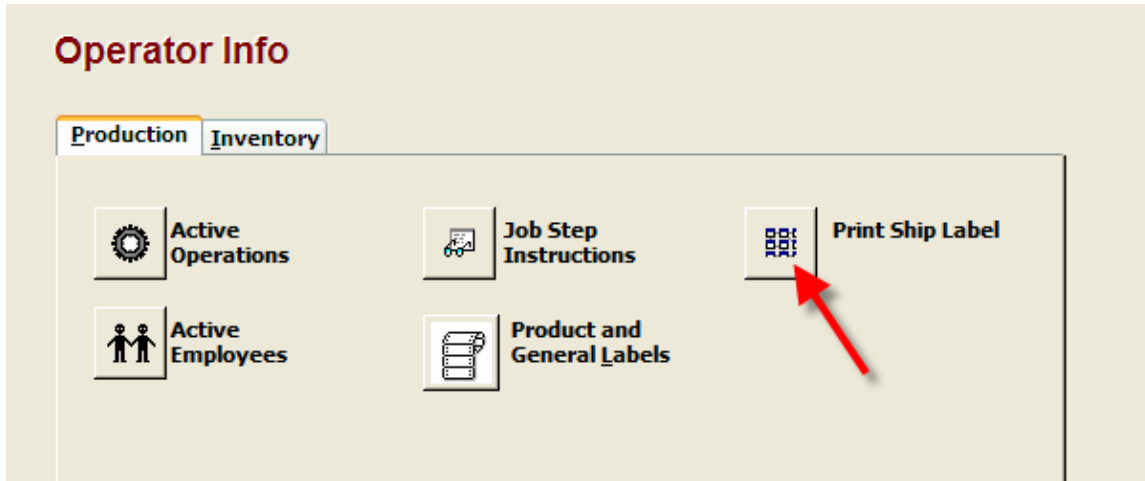
Tag serves as the interface to take the material tracking information in the BellHawk® database and to automatically pass this information to BarTender® Enterprise to produce the required labels with minimal operator intervention.

Tag can also retrieve the data encoded in tracking barcodes and RFID tags as part of the label printing process and store these back in the BellHawk database. This is so that this tracking information can be used by the BellHawk material tracking software and also so it can be used to automatically generate Advanced Shipment Notice (ASN) data submissions including DoD Wide Area Work Flow (WAWF) submissions.

## Operator Interface

### Printing Shipping Labels

To Produce a Shipping Label, go to the Operator switchboard and select the Print Ship Label icon from the Inventory tab:



This will bring up the following screen:



Scan the tracking barcode on the container (such as a pallet) to be shipped, select the label format and then select the Print button and the label will be automatically printed. Please note that the container barcode must be for a container that has been picked for an order or for a container (such as a pallet) into or onto which materials have been picked for an order.

If different shipping labels are required for different containers or different customers, then the label format can be specified as a user attribute for the container type or the customer. In this case the designated label format name will be automatically displayed for the operator approval.

This can be repeated for as many shipping containers as needed, simply by scanning their tracking barcodes and selecting print.

An example commercial shipping label produced from BellHawk is shown below:

<b>Ship From</b>
BellHawk Systems Corporation 45 River Street Millbury MA 1527, USA
<b>Ship To</b>
XYZ Distributing 123 Main Street Anytown ME 3126, USA

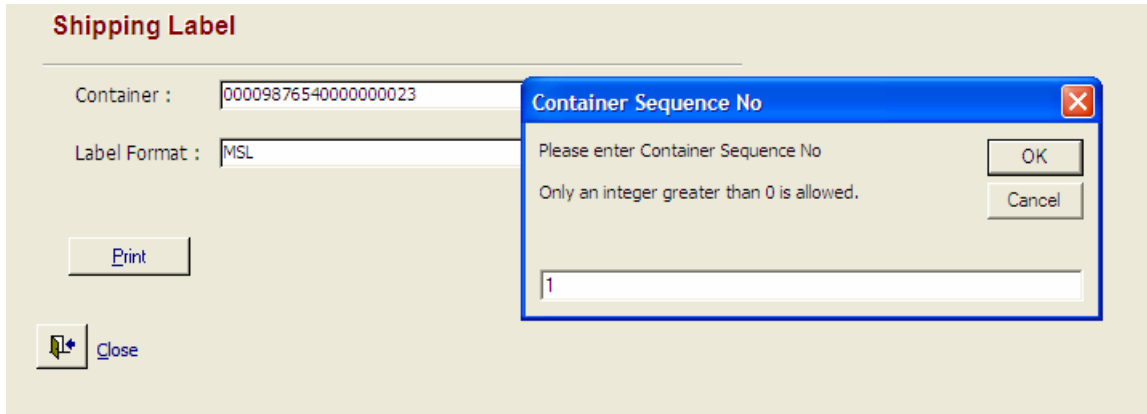
(00) 0 0987654 000000002 3

All the data on the label comes directly from the BellHawk database. This enables one standard format label to print shipping labels for many customers with minimal operator input. This also minimizes the opportunities for operator error.

The BellHawk Tag module automatically supplies a unique serial number for each shipping label, as shown in the above example with a UCC 128 SSCC barcode.

If the SSCC code has been designated as the tracking barcode in the label this barcode will now become the tracking barcode for this label in BellHawk.

Labels can also contain fields that are requested at time of label printing:



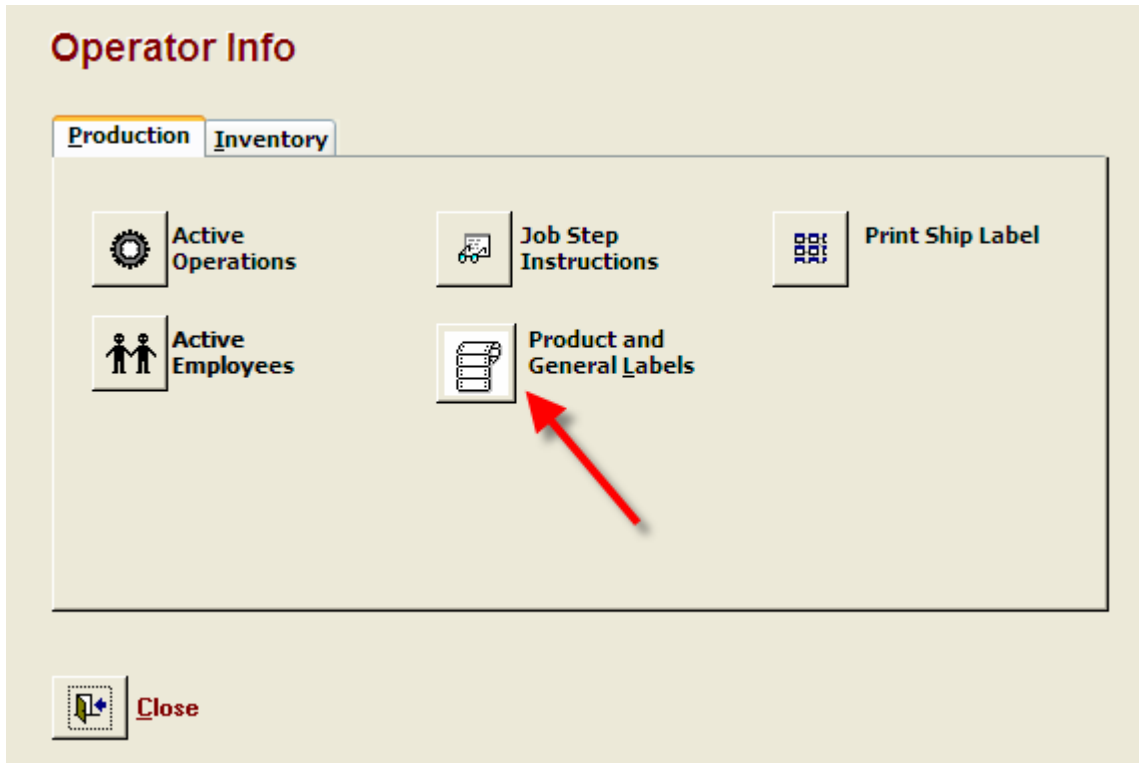
This is often required in complex labels, such as this DoD Military Shipping Label printed directly from BellHawk:



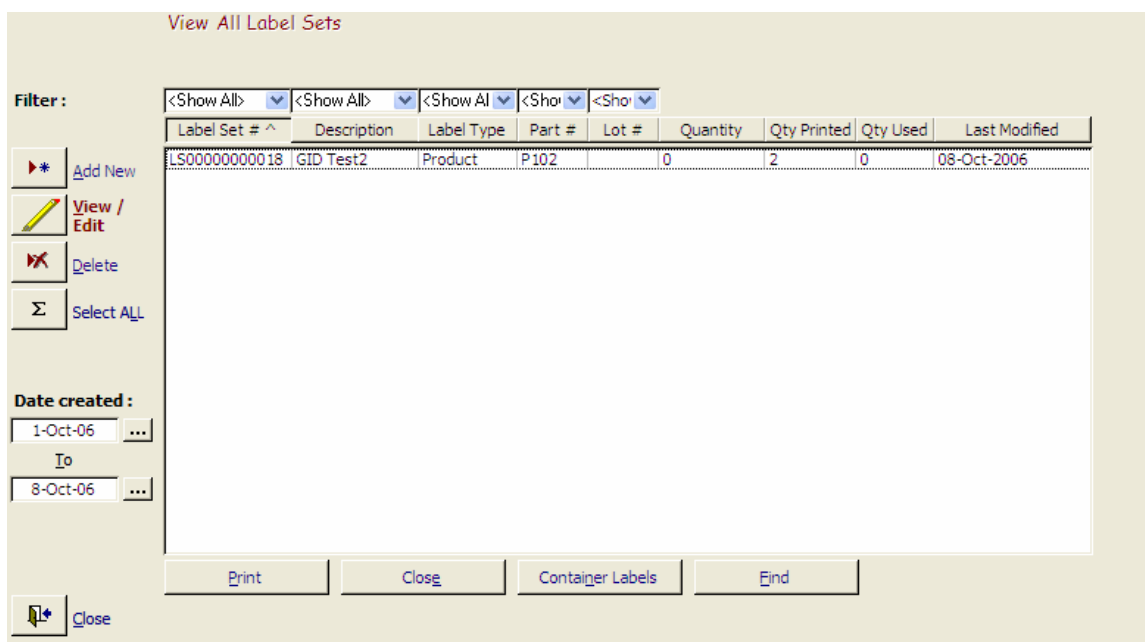
The contents of the TCN and of the PDF417 barcode are setup directly out of the BellHawk database as is the contents of the embedded RFID tag. The encoded contents of the TCN barcode and the RFID tag are then read back into BellHawk for tracking purposes.

### Printing Product Labels

To Print a Product Label go to the Operator switchboard and select the Product and General Labels icon:



This will bring the user to an All screen on which all previously produced sets of Product, Product Box and General Labels are displayed:



Tag enables user to track the sets of labels they produce as they are used. Once all the labels in a set are consumed they will no longer appear in this list. That is, unless a date range when created is selected, to look at prior label sets. The label sets displayed can be filtered by Label Type, Part Number and Lot Number. Also wild cards can be used to search in the label set descriptions to find a specific label set.

To create a new label set, click on Add/New and this will bring up the detail screen for the label set:

**Add New Label Set**

LabelSet Number # :

Label Type :

Label Set Description :

Part Number :  ...

Label Name :

Lot Number :

Quantity :

Container Type :  ...

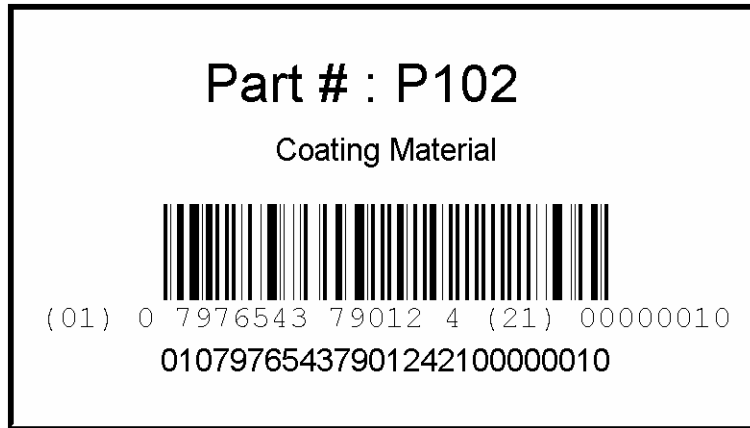
Default No of Labels to print :

Serial Numbers    Print    Container Labels    Close

On this screen, the user will be asked to select or enter:

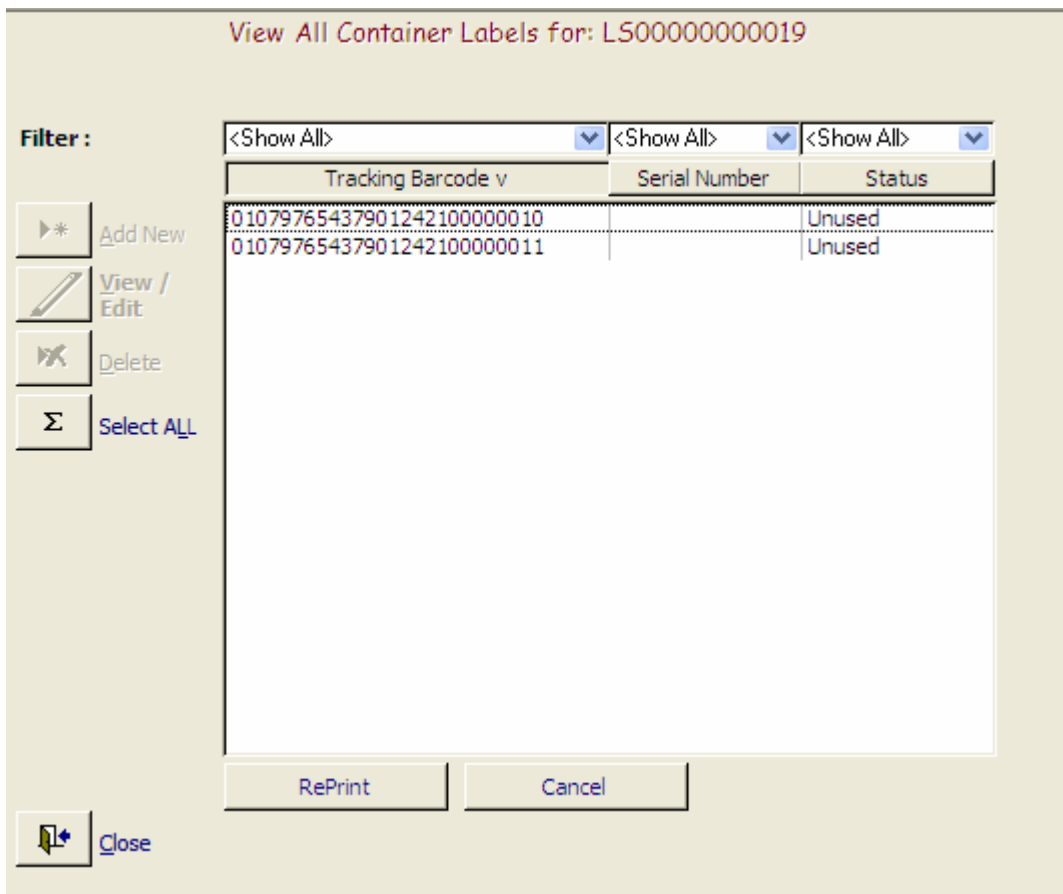
1. The label type (Product, Product Box, or General Box label)
2. A description for this label set (to assist in tracking the label sets).
3. The part number. This can be typed in, selected from a drop-down list, or the ellipses selected to go to the Item Master Parts All screen in selector mode.
4. The label name. This is the short-form name stored in BellHawk, which references a BarTender label format files (see Label Setups).
5. The lot number for the parts to which this set of labels will be applied. This is only requested if the Part Number is designated as being lot tracked in the part setup.
6. The Quantity in the container. This is grayed out for individually barcoded items.
7. The Container Type, this is grayed out for non-serialized items.
8. The default number of labels to be printed in a set.

When this data has been entered then the user is able to print a set of labels by selecting the Print button. This will cause the labels to be printed on the default printer. Prior to printing, the user will be asked for the number of labels to print and then these labels will be printed:



This example label has a serialized GTIN barcode label using concatenated GIAI standard codes. The resultant tracking barcode (as produced by a scanner scanning the tracking barcode) is shown at the bottom. This is automatically returned to BellHawk and entered as the tracking barcode associated with the label.

The user can then select the Container Labels Button on the detail screen to see a list of labels in this label set:



If needed, one or more labels can be selected from the label set and reprinted.

From this you can see that the contents of the Serialized GTIN barcode created by BarTender has been transferred back to BellHawk as the tracking barcode associated with each of these labels.

The labels so created can then be scanned as the tracking barcode for a product, to which they are then attached, during the following transactions:

1. Enter Material into Inventory.
2. Receive against Vendor Purchase Order
3. Simple Receive
4. Material Out from Job Step
5. Return Material from Job Step.

In each case, the following parameters associated with the label are automatically transferred to the container to which it is attached:

1. Part Number
2. Serial Number (if serialized)
3. Lot Number (if lot numbers are being tracked)
4. Quantity
5. Container Type

As each label is used, the system will track its use, including the product to which it was attached. The quantity-printed and the quantity-used are reported on the All screen for label sets.

The View/Edit button on the All screen can be used to return to the detail screen for a selected label set.

**Edit Label Set**

LabelSet Number # : LS00000000019

Label Type : Product Label

Label Set Description : GTIN Set 3

Part Number : P102

Label Name : GTIN

Lot Number : G524357

Quantity :

Container Type :

Default No of Labels to print : 2

Serial Numbers    Print

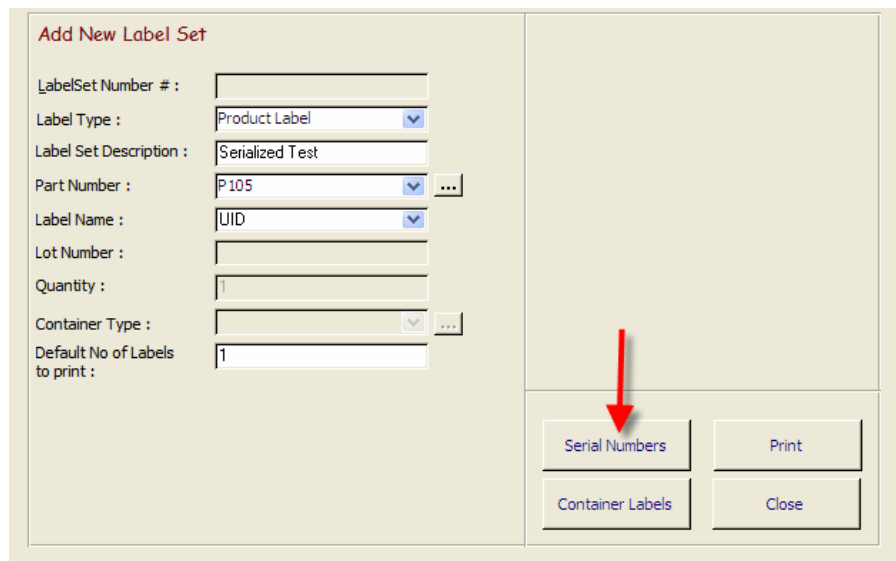
Container Labels    Close

The only fields that can be edited are the label set description, the label name (to be used for subsequent labels printed in this set) and the default no of labels (for subsequent labels printed). This screen can be used to print more labels from the set and to reprint container labels through the Container Labels button.

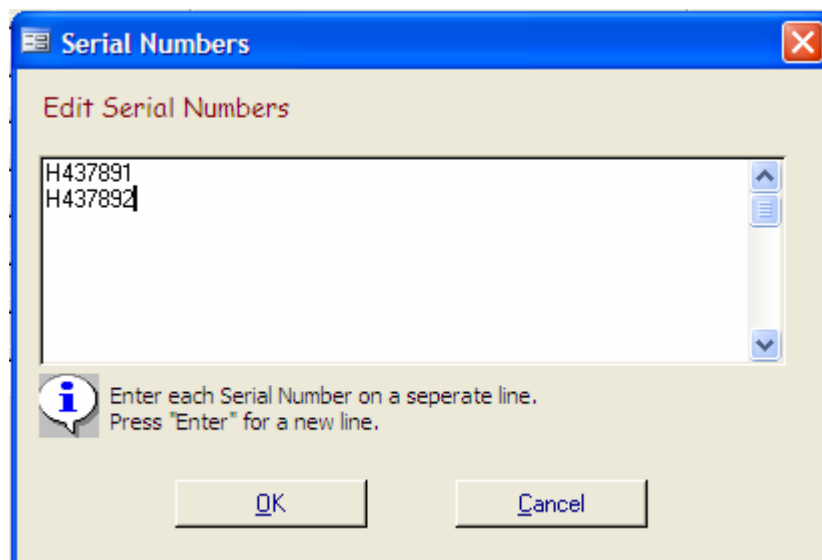
The Close button on this detail screen and on the label set All screen will Close the label set so that it can no longer be used to print labels, Also it will disappear from the All screen after a designated number of days.

### **Printing Product Labels with Serial Numbers**

The printing of product labels with serial numbers is almost identical to printing labels for non-serialized products except that, after entering the data about the label set, the Serial Number button is selected to bring up a screen in which to enter serial numbers:



The screenshot shows the 'Add New Label Set' dialog box. It contains several input fields: 'LabelSet Number #', 'Label Type' (set to 'Product Label'), 'Label Set Description' (set to 'Serialized Test'), 'Part Number' (set to 'P105'), 'Label Name' (set to 'UID'), 'Lot Number', 'Quantity' (set to '1'), 'Container Type', and 'Default No of Labels to print' (set to '1'). At the bottom right, there are four buttons: 'Serial Numbers', 'Print', 'Container Labels', and 'Close'. A red arrow points to the 'Serial Numbers' button.



The screenshot shows the 'Serial Numbers' dialog box. The title bar reads 'Serial Numbers'. The main area is titled 'Edit Serial Numbers' and contains a list box with two entries: 'H437891' and 'H437892'. Below the list box is an information icon and the text: 'Enter each Serial Number on a separate line. Press "Enter" for a new line.' At the bottom, there are 'OK' and 'Cancel' buttons.

These serial numbers are placed in a “pool” and used in sequence as serialized labels are printed. Whenever the Serial Numbers button is pressed, the serial numbers remaining in the pool for this label set are displayed. New serial numbers can be added to a label set whenever needed in order to print more labels. The serial numbers can be cut and pasted into the serial number box from an external source.

These serial numbers can then be printed on the label and used to form tracking barcodes, such as the UID 2D DataMatrix label shown on the following DoD UID label:

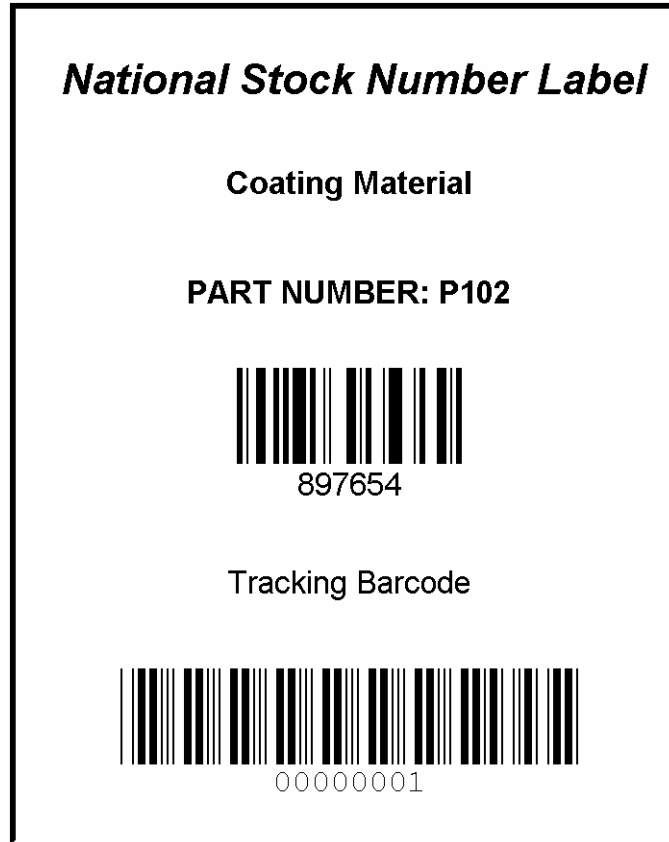


This label has the DoD required 2D barcode containing the company’s CAGE (Government identifier) code, the part number and the Serial Number (entered in the Serial Number box). It also has a linear barcode containing the UII information that a 2D barcode scanner would deliver after scanning the 2D barcode. This enables less-expensive 1D barcode scanners to be used to track the same item with the same tracking barcode. This tracking information is stored back in BellHawk as the tracking barcode for the item to which the barcode label is attached.

### ***Printing Product Container Barcodes***

This is essentially identical to printing product containers. By separating the product container barcodes into a separate set of barcodes we minimize the possibility that the users mixing up the label formats for products and product boxes. We also make it easier to setup default labels for these two cases. Thus we might set up a UID label for a product but setup a Product Box label with a National Stock Number barcode and an RFID tag for tracking purposes.

An example of a National Stock Number barcode produced from BellHawk is shown below:



This barcode has the National Stock Number (NSN) which came from a user defined attribute NSN on the part. It also has a unique serialized tracking barcode that was automatically generated by BellHawk. Please note that there can be multiple user stations generating the tracking barcodes and, because the unique numbers are issued by the central BellHawk server, they are all unique.

### ***Printing General Container Barcodes***

These proceed pretty much as with producing Product Container labels except that there is far less specific information available from within the BellHawk database for these labels. An example of printing out general container labels is printing BellTags (rolls of pre-printed license plate tags), an example of which is shown below:



This label contains a unique tracking barcode that is produced from the same sequence as those described for the Product Container barcodes (so they will not be confused). The tracking barcode number is saved back in BellHawk so that it can be seen in the set of labels.

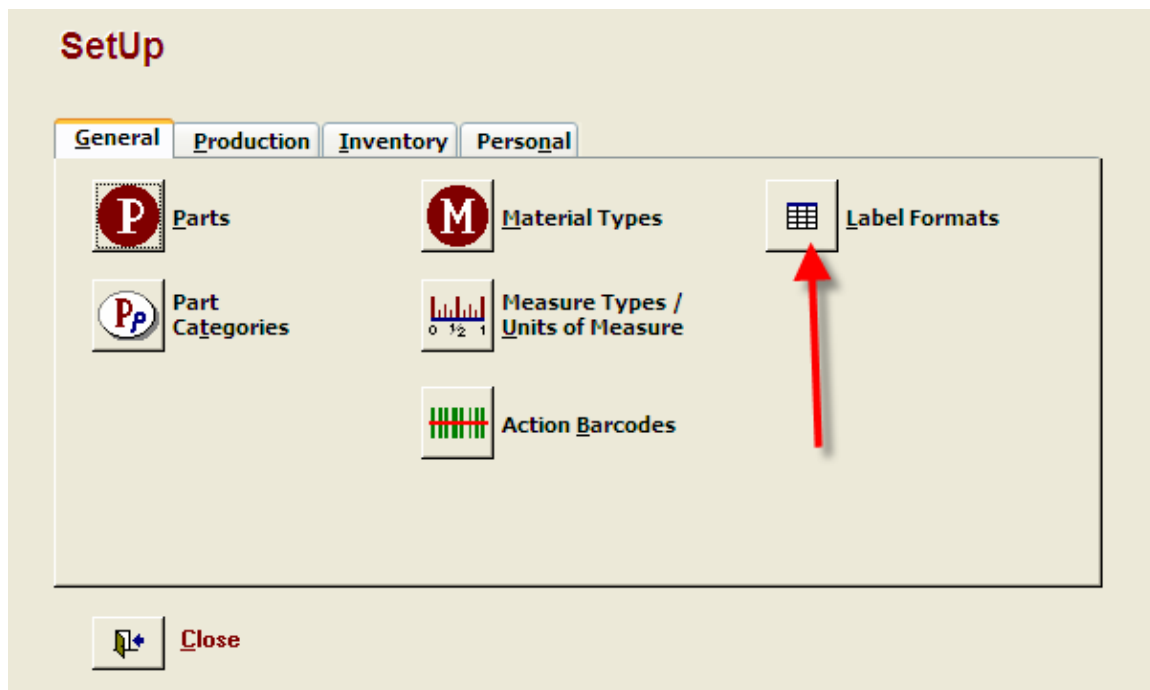
## Setting-Up Label Formats

### *Introduction*

This section presumes that users are familiar with how to use the label creation capabilities of BarTender Enterprise and also are familiar with the Object Attribute setup and Import/Export mechanisms in BellHawk.

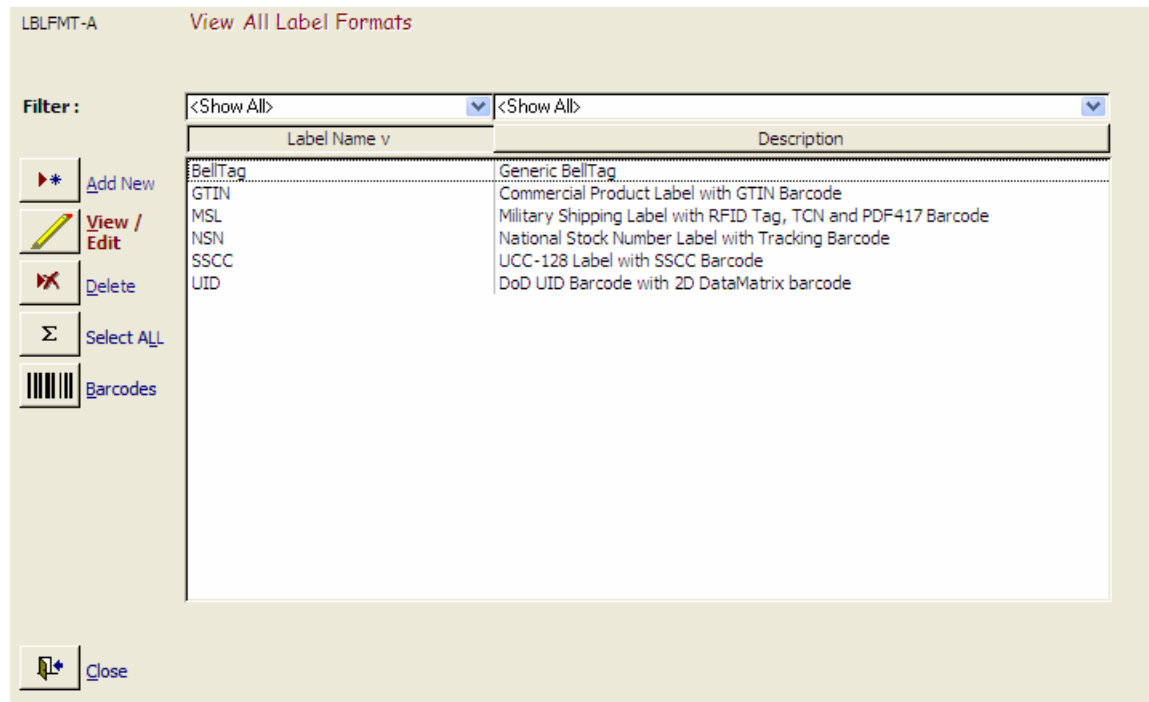
### *Label Names*

On the Setup switchboard, there is a Label Formats button:



This leads to a set of screens that enable the BellHawk internal label names to be associated with the BarTender Label format names.

Selecting the Label Formats button will lead to a Label Formats All screen:



From this screen, label formats setups can be added or edited. The Label Formats detail screen (reached from Add/New or View/Edit is as follows:

The various data entry fields are as follows:

1. Label Name: This is the internal BellHawk name by which the label is referenced by operator printing labels. It must be unique.
2. Label Format File is the full path the corresponding BarTender label format. If the BarTender label formats are kept in a directory on a server, then the server name should be referenced in the path so that the files are retrieved from the server and not the local computer. The ellipses

bring up a directory selection control. Just remember to first select MyNetwork places and the server name in setting up the directory path.

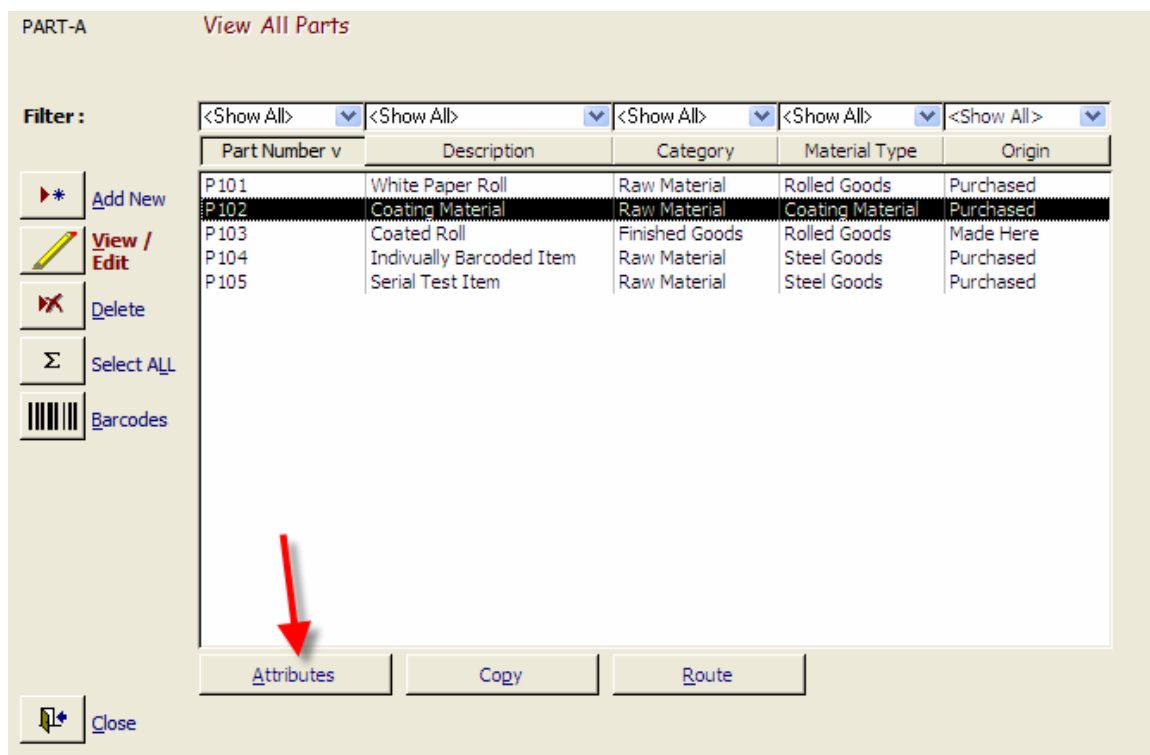
3. Description: This is a description to assist the operator in selecting the appropriate label.
4. Product/Product Container/General/Ship selects the category in which the label falls.
5. Is Debug: If selected prints out a report, prior to printing the label, of all the attribute values used in printing the label and of all attribute values available for printing the label but not used. This is very useful for debugging label formats.

## Field Name Association

The basic principle on which the Tag module works is to associate an attribute name in BellHawk with a field name in BarTender. When the label is printed from BellHawk the value of the attribute is automatically entered into the BarTender field.

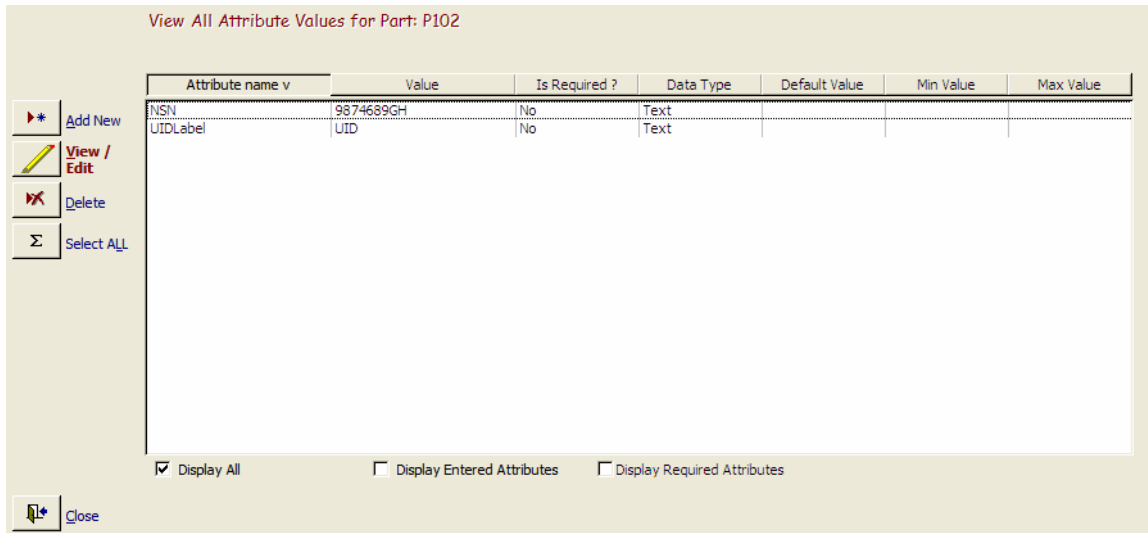
Let us look at the specific example of the NSN attribute (used in our National Stock Numbered Product Box label above).

In BellHawk we define a user attribute for a Part by selecting the Attributes button on the Item Master All or Detail Screens.:



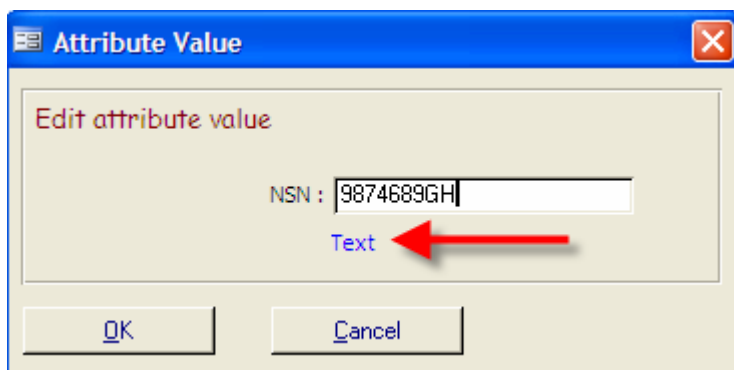
You will find these Attribute buttons associated with many Objects in BellHawk. They are used to specify the user attributes for the object.

Selecting the Attributes button for a selected object instance, leads to an Attributes All Screen:



This shows all the available attributes for that object along with the values for those with defined values.

To set or change the attribute value, simply double click on the attribute (or select the attribute and click on View/Edit) and this brings up the following screen:



Here the value of the attribute can be set or edited. The type of the attribute (Text, Integer, Float, Symbolic) is indicated below the data entry box. The entry will be checked against this for correct data formatting.

To Add a new attribute name, click on the Add New button on the View All Attributes screen and this will bring up a screen to create new attributes:

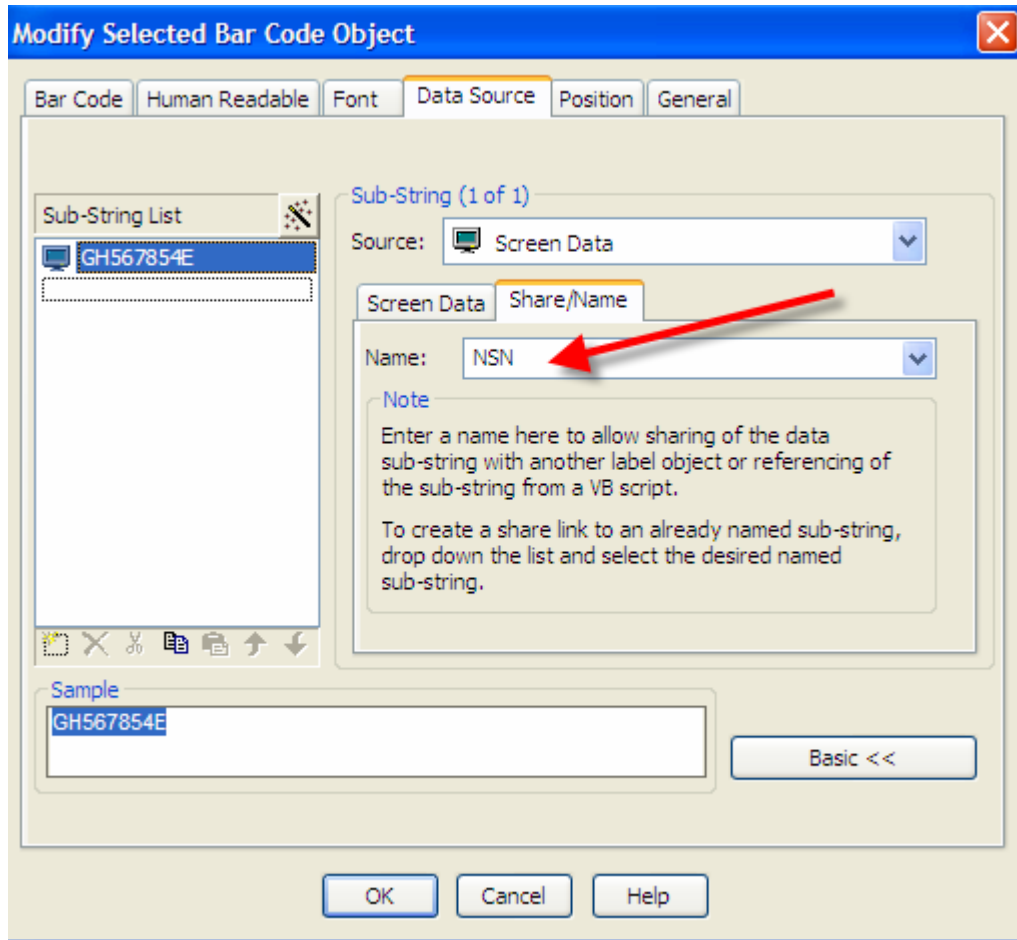
The screenshot shows a web form titled "Add New Attribute for: Part". It contains several input fields: "Attribute Name" (text box), "Attribute UOI" (text box), "Attribute DataType" (dropdown menu), "Attribute Type" (dropdown menu), "Select Table" (dropdown menu), "Select Field" (dropdown menu), and "Default Value" (text box).

Details of how to use this screen are described in the Object-Attribute section of the BellHawk Users Manual. The setup for our example NSN attribute is shown below:

The screenshot shows a web form titled "Edit Attribute for: Part". The fields are populated with the following values: "Attribute Name" is "NSN", "Attribute UOI" is "Part.NSN", "Attribute DataType" is "Text", "Attribute Type" is "Optional", and "Default Value" is empty.

Now we have setup the attribute value for our Part P102, that is all we have to do on the BellHawk end to have this attribute value available for printing on the BarTender label.

In BarTender we setup the same name NSN as a field name in BarTender:



This name relation ties the BellHawk attribute name to the BarTender Share/Name for a field. This field name can then be used as text on a label, as the contents of a barcode on the label or as a sub-field in a composite data field.

When a label is printed, BellHawk reads back all the field names from the specified BarTender label format and enters the attribute values in the specified fields. If it finds a field name that it does not have an attribute for then it asks the user for the value of the field name and sets it as a temporary attribute value. (Note do not set a named field as a prompt field in BarTender otherwise you will be asked for the value twice – once by BarTender and once by BellHawk).

Once it has setup all the named fields then BellHawk will command BarTender to print the label on the printer designated in the BarTender label format.

### **System Attributes**

As well as the user defined attributes, there are many system attributes such as ShipToName and ShipToAddress that are defined internally by BellHawk. These are listed at the end of this Chapter of the User's manual. Different system attributes are available depending on the circumstances. The attribute PartNumber is available for printing on Product and Product

Container labels but not for General Container labels, for example. The user does not have to define these attributes but may do so, if they wish to over-ride the default values in a specific case.

Please note that all Attribute Names and Field Names are case sensitive.

### **Control Attributes**

There are certain attributes that are used to control the actions of printing the label rather than used as a field name in the label. The currently defined control attributes are:

- **MSL** – must be set to the text string "Yes" otherwise the TCN and PDF417 in a military shipping label will not be generated or listed in the attribute value report.
- **Track** – if Track is set to the string "No" then the tracking barcode is not scanned or retrieved from the label at time of label production and the individual labels are not tracked. This is used for producing UPC or GTIN labels for products that do not have tracking barcodes.

### **iCodes**

iCodes are pseudo Attributes that generate special strings to be inserted into the BarTender labels. They are referenced by setting the BarTender field name to the iCode name. Some iCodes depend on user defined attribute values and will only be generated if the user attribute values are available or set to specific values. Most iCodes are inputs to the label format but three, iTrack, iUID and iRFID are special in that the BarTender fields referenced by these are automatically loaded into the label or containers tracking barcode, UID/UII and RFID fields. This eliminates the need to subsequently scan these after creating the label.

The following is a table of iCodes:

<b>iCode</b>	<b>Purpose</b>
iSerialNo	Next user provided Serial Number from pool. If SerialNumber appears multiple times then only one serial number is used per label.
iItemSeqNo	This is the Item Sequence Number. It is incremented for every printed label it is used on. It can be used to provide a unique serial number for a part. Note that it is unique over all parts and not within a part.
iPartSerialNo	This is a serial number generated sequentially by BellHawk for individual parts. It is incremented by 1 every time it is used. The initial value can be set on the handling tab of the item master parts setup screen.
iBarcodeSeqNo	This is a unique number and is incremented from each label it is printed on. It is used to place a unique tracking barcode on container labels,

iShippingLabelSeqNo	Provides a unique serialized number for each shipping label that is printed. This can be used as a field in a UCC 128 SSCC barcode.
iTCN	Generates a TCN Barcode for a MSL based on user supplied and system attribute values (see section on Generating MSLs) if MSL = "Yes"
iPDF417	Generates a PDF417 Barcode for a MSL based on user supplied and system attribute values if MSL= "Yes"
iTrack	Returns tracking barcode from the label unless Track = "No"
iUID	Returns the decoded UID/UII code from the label – this is what is reported to WAWF – should not contain control characters
iRFID	Returns encoded contents of RFID label – this is what is reported to WAWF

The initial values for iItemSeqNo, iBarcodeSeqNo and iShippingLabelSeqNo are set on the Admin switchboard, Sys Admin |BellHawk Parameters | Host Info tab. These are incremented by 1 every time they are used as is iPartSerialNo. Note that these numbers are kept in the database so that a unique number is issued to every user of the system so there is no duplication of tracking numbers.

### **Setting Up Returned iCodes in BarTender**

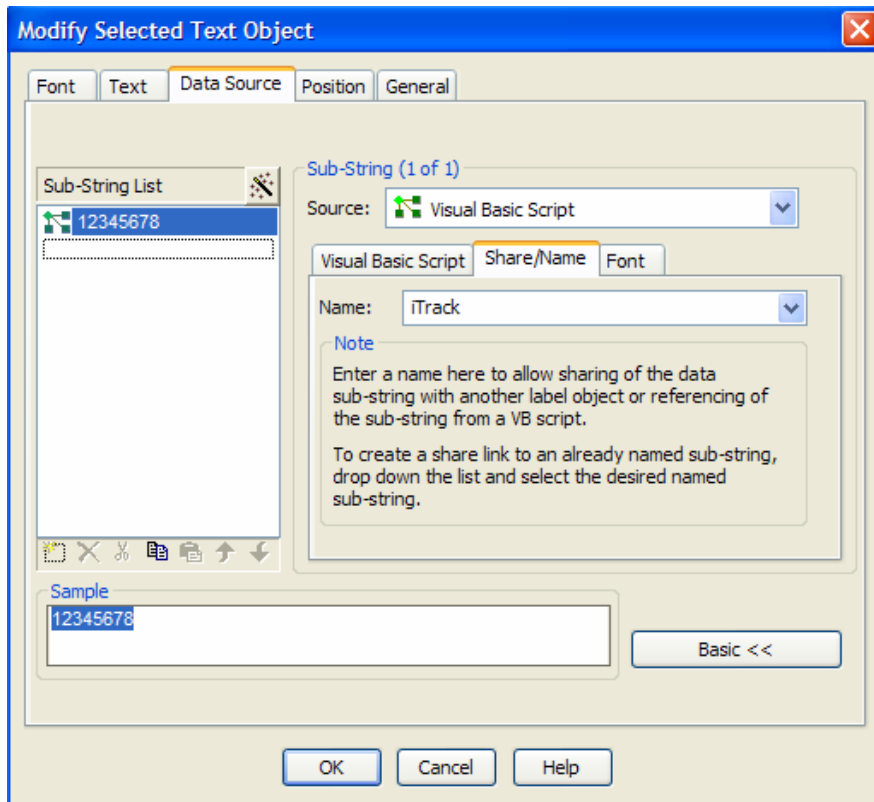
The purpose of returning tracking barcodes after printing a label is to avoid the need for the user to scan the tracking barcode or RFID tag after generating the label. When it prints a label, BellHawk has no way of knowing which barcode is the tracking barcode. Many tracking barcodes are made up from multiple subfields and may have checksum digits appended that will appear as part of the barcode string when the label is printed.

When BellHawk has finished populating a label it reads back the contents of the fields iTrack and iRFID and stores these with the label/container record. This is so they can be reported in an ASN/WAWF submission and also used in tracking the container to which the label is attached.

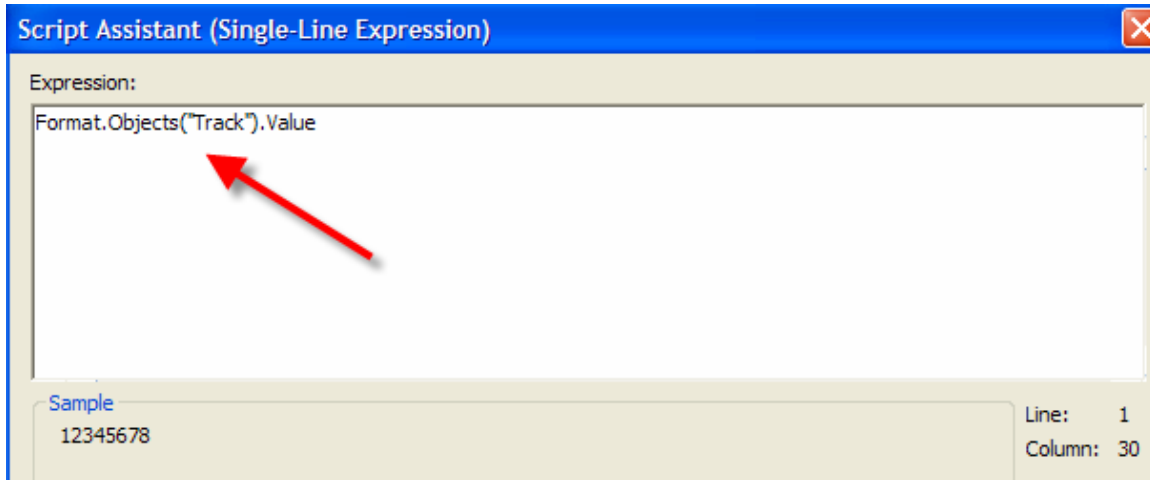
Let us start with the simplest case, that of returning the tracking barcode from a BellTag:



To do this we setup a non-printing text field (pointed to by arrow above) and give it a field name iTrack in BarTender:



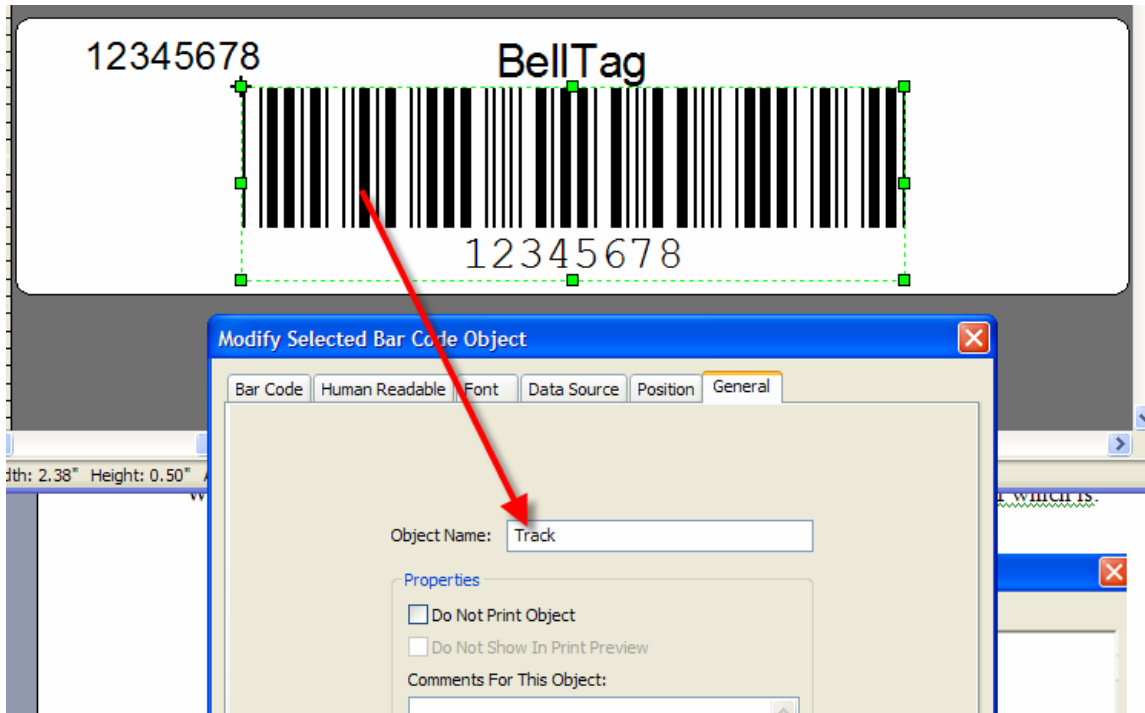
We also specify that the Source of the Data is a Visual Basic Script, the contents of which is:



The Format.Objects( ).Value function returns the text string that comprises the data object whose name is referenced in the call, in this case "Track".

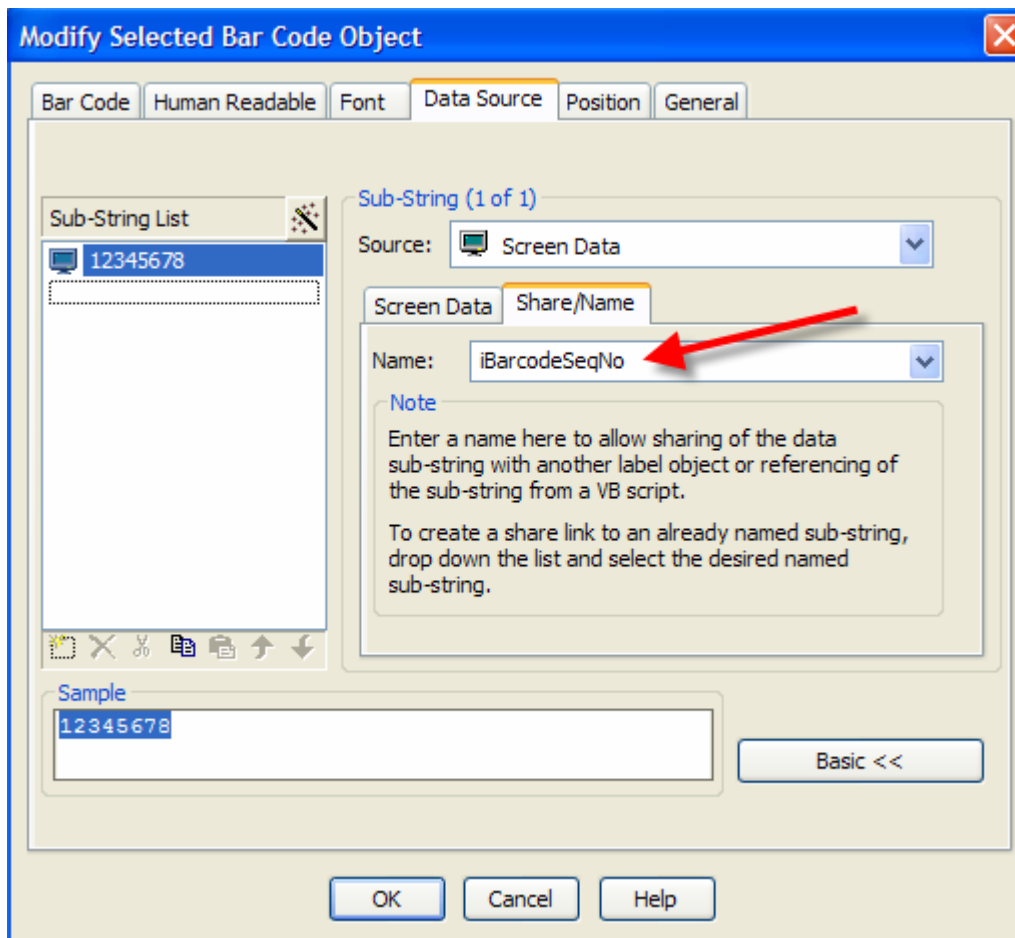
This name references the name set on the General Tab of the object (in this case the tracking barcode) whose value we wish to return in iTrack:

Note that the general Tab (for our iTrack text string) is where we also specify for our iTrack



string, that it is non-printing.

In this example the barcode on the BellTag comes from our IBarcodeSeqNo iCode:



Which will be returned as the tracking barcode for the label in iTrack and placed in the tracking barcode for the container/label.

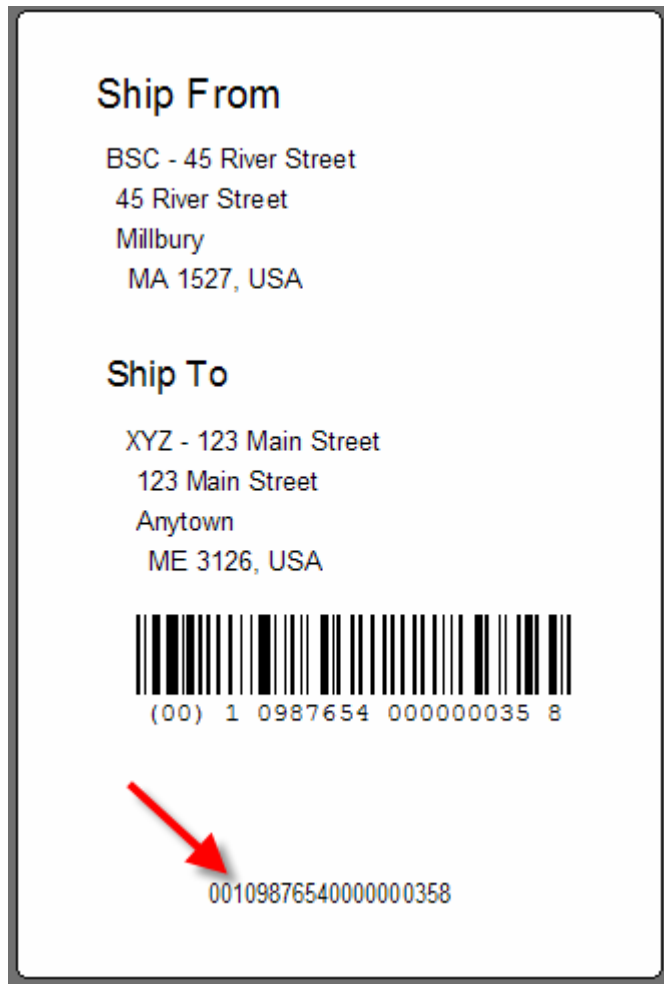
You might ask why we would go to so much trouble, if we know the tracking barcode inside BellHawk at the outset. The person setting up the BellTag label could have included a prefix as a screen text field before the barcode sequence number to form a more unique barcode. In this case, the prefix concatenated with the barcode sequence number would be returned in iTrack.

We can use this method to return the TCN code from a Military Shipping Label as the tracking barcode.

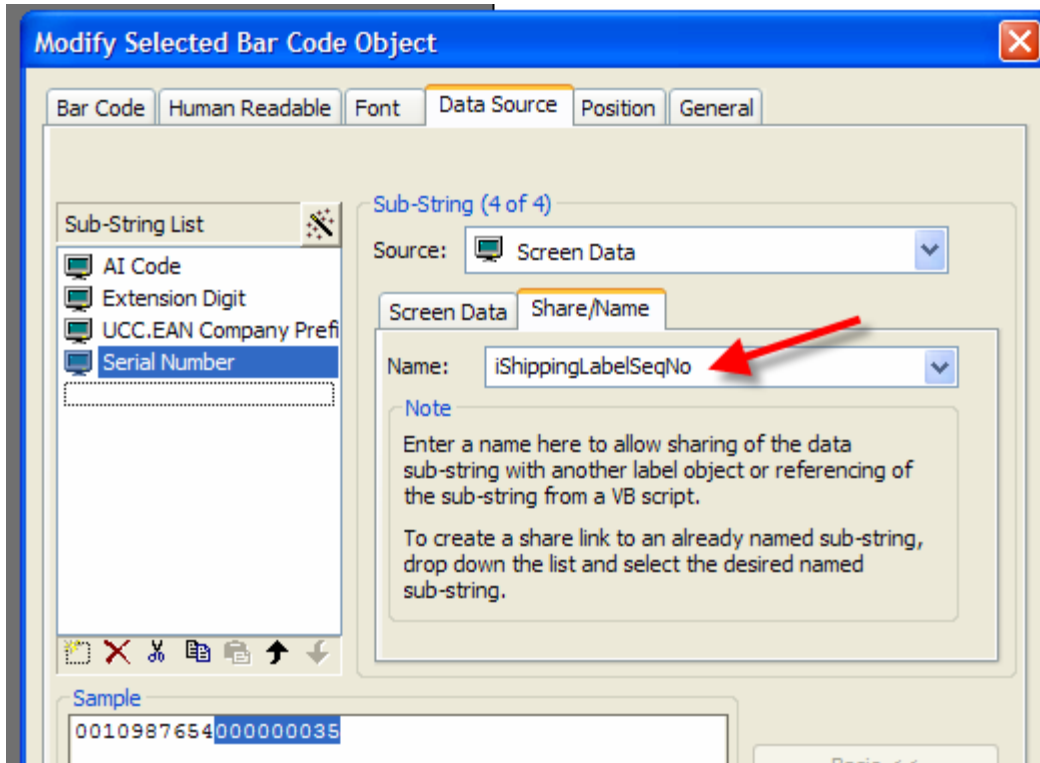
We can use a similar method to return iUID. This is the decoded value of the UID 2D barcode, which is what you will get when you scan the 2D barcode with a scanner that strips out all the control and header characters. This is often referred to as the decoded UII or simply the UII and is what is reported to WAWF as part of the ASN when the unit is shipped to the DoD. With the UID code, we do not directly reference the 2D barcode as this contains control characters. Instead we separately form the UII as a non-printing string on the label using the same UID format as the DataMatrix barcode and then reference this as the source object for iUID.

It gets a little trickier dealing with GIAI standard barcodes as some of these have an additional checksum digit that will appear as part of the barcode when it is scanned. Let us consider the case of a commercial shipping label:

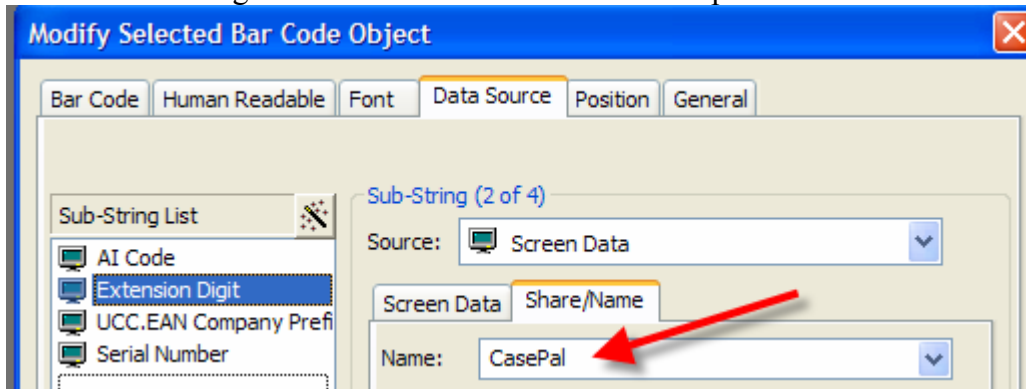
Here we have the same non printable iTrack field but its formation is made a little more complex by the checksum digit in this case the last 8 that is added by BarTender (according to the GIAI standards) in forming the SSCC code.



The SSCC barcode is a UCC/EAN Code 128 (set on the Bar Code tab) and is made up of a number of sub-fields:

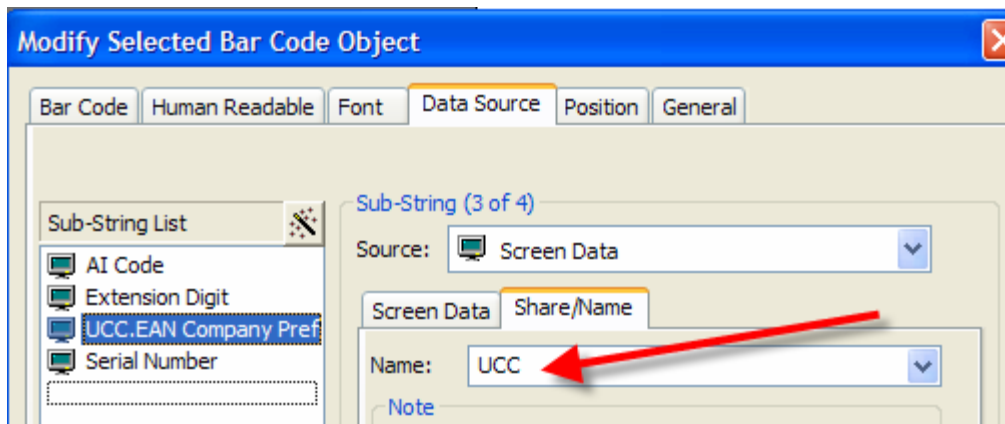


We see here the use of the iShippingLabelSeqNo for the unique serial number for the container. The Extension Digit defines whether this is a case or a pallet:

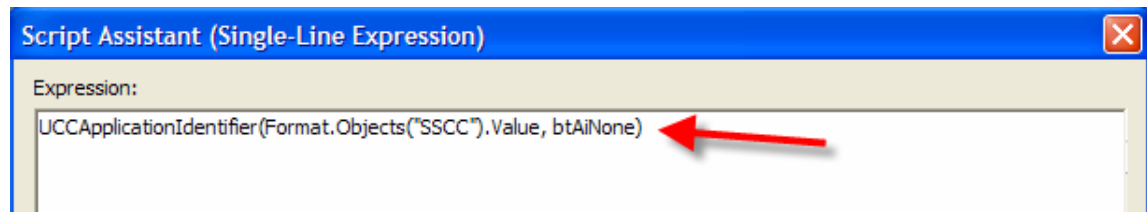


CasePal is a User Attribute defined for Containers and is 0 for Cases and 1 for Pallets.

The UCC.EAN Company Prefix is set by the User Attribute UCC, which is usually defined at the level of the DBA for which the shipment is being made:

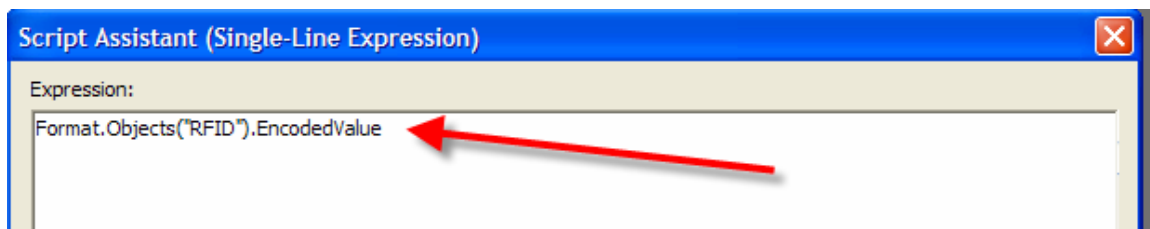


If we simply used a Visual Basic Script: `Format.Object("SSCC").Value` for our iTrack string (where the SSCC barcode object is named "SSCC" in BarTender) then we would get back the string value without the last check digit, as BarTender regards this as the value of the object "SSCC". To get the value of the object with the check digit (which is what we get when we scan the SSCC barcode), then we need to use the following as the Visual Basic expression for our iTrack field:



For GIAI codes this will format the sting, returned from `Format.Objects("SSCC").Value` by adding the checksum. There are other formatting options available, based on the last parameter but this is the one to use for iTrack returns.

For returning the contents of an RFID tag we use a similar principle and form a non-printing string with a field named iRFID. The value encoded in the RFID tag is then returned using the Visual Basic script:



For the iRFID field. The returned value is stored in the RFID field in the Label/Container so it can be used in exit portal tracking and also reported using WAWF and in other ASN submissions.

Please note that the `Format.Objects("RFID").EncodedValue` returns the string "<Empty>" if the selected printer for the label is not an RFID printer.

### System Attribute Values Available

Note that we have also included some commonly used User Defined attributes for reference

#### ***For Product , Product Box and General Container Labels***

##### From Container Type Records

Object Name	Attribute Name	System or User	Comes From/Description
Container	ContainerType	S	tblContainerTypes.ContainerType
Container	TareWeight	S	tblContainerTypes.TareWeight
Container	TareWeightUOM	S	From tblContainerTypes.TareUOM=> tblUnitsOfMeasure.UnitName
Container	TareWeightUOMSF	S	From tblContainerTypes.TareUOM=> tblUnitsOfMeasure.UnitNotation
Container	CasePal	U	0=Case; 1 = Pallet

##### From Item Master Records

Not available for General Container Labels.

Object Name	Attribute Name	System or User	Comes From/Description
ItemMaster	PartNumber	S	tblItems.ItemNumber – Local Part Number
ItemMaster	PartDescription	S	tblItems.ItemDescription
ItemMaster	ItemGID	S	tblItems.ItemUOI – Global Identifier
ItemMaster	PrimaryUnitCost	S	tblItems.UnitCost
ItemMaster	PrimaryUnitPrice	S	tblItems.UnitPrice
ItemMaster	SecondaryUnitCost	S	tblItems.UnitCost divided by UOMConversionFactor
ItemMaster	SecondaryUnitPrice	S	tblItems.UnitPrice divided by UOMConversionFactor
ItemMaster	PrimaryMeasureType	S	tblItems.PrimaryUOMTypeID=> tblMeasureTypes.MeasureType
ItemMaster	PrimaryMeasureTypeSF	S	tblItems.PrimaryUOMTypeID=> tblMeasureTypes.mtCode
ItemMaster	SecondaryMeasureType	S	tblItems.SecondaryUOMTypeID=> tblMeasureTypes.MeasureType

ItemMaster	SecondaryMeasureTypeSF	S	tblItems.SecondaryUOMTypeID=> tblMeasureTypes.mtCode
ItemMaster	PrimaryQuantityUOM	S	tblItems.PrimaryUOMID=> tblUnitsOfMeasure.UnitName
ItemMaster	PrimaryQuantityUOMSF	S	tblItems.PrimaryUOMID=> tblUnitsOfMeasure.UnitNotation
ItemMaster	SecondaryQuantityUOM	S	tblItems.SecondaryUOMID=> tblUnitsOfMeasure.UnitName
ItemMaster	SecondaryQuantityUOMSF	S	tblItems.SecondaryUOMID=> tblUnitsOfMeasure.UnitNotation
ItemMaster	PartCategory	S	tblItems.CategoryID=> tblItemCategories.CategoryDescription
ItemMaster	PartMaterial	S	tblItems.MaterialID=> tblMaterials.MaterialName
ItemMaster	UnitNetWeight	S	tblItems.NetWeight
ItemMaster	PartDBA	U	Is the DBA for which part is being produced
ItemMaster	UPC	U	Universal Product Code for part
ItemMaster	NSN	U	National Stock Number for Part
ItemMaster	ProductLabel	U	Commercial Product Label for Part
ItemMaster	ProductCaseLabel	U	Commercial Product Container Label for Part - may hold multiple parts
ItemMaster	UIDLabel	U	DoD UID Label
ItemMaster	NSNLabel	U	DoD Label for Product Container

### From DBA Object

Note DBA instance defaults to Host DBA unless specifically set by attribute PartDBA for Item Master.

Object Name	Attribute Name	System or User	Comes From/Description
DBA	DBAName	S	tblDoBusAs.DoBusAsName
DBA	DBAStreet	S	tblDoBusAsAddress.StreetAddress where IsPrimaryShipFrom =1 and DoBusAsID = tblDoBusAs.DoBusAsID
DBA	DBAStreet2	S	tblDoBusAsAddress.StreetAddress2 where IsPrimaryShipFrom =1 and DoBusAsID = tblDoBusAs.DoBusAsID
DBA	DBACity	S	tblDoBusAsAddress.City where IsPrimaryShipFrom =1 and DoBusAsID = tblDoBusAs.DoBusAsID

DBA	DBAState	S	tblDoBusAsAddress.State where IsPrimaryShipFrom =1 and DoBusAsID = tblDoBusAs.DoBusAsID
DBA	DBAZIP	S	tblDoBusAsAddress.ZipCode where IsPrimaryShipFrom =1 and DoBusAsID = tblDoBusAs.DoBusAsID
DBA	DBACountry	S	tblDoBusAsAddress.Country where IsPrimaryShipFrom =1 and DoBusAsID = tblDoBusAs.DoBusAsID
DBA	DBAContact	S	tblDoBusAsAddress.ContactName where IsPrimaryShipFrom =1 and DoBusAsID = tblDoBusAs.DoBusAsID
DBA	DBAPhone	S	tblDoBusAsAddress.Phone where IsPrimaryShipFrom =1 and DoBusAsID = tblDoBusAs.DoBusAsID
DBA	DBAFAX	S	tblDoBusAsAddress.FAX where IsPrimaryShipFrom =1 and DoBusAsID = tblDoBusAs.DoBusAsID
DBA	UCC	U	DBA UCC Code - Company Prefix for UPC labels and GTIN labels and SSCC labels
DBA	CAGE	U	DoD Cage Code for DBA

### From Host DBA Object

Object Name	Attribute Name	System or User	Comes From/Description
Host	HostName	S	tblParameters.HostDoBusAsID=> tblDoBusAs.DoBusAsName
Host	HostStreet	S	tblDoBusAsAddress.StreetAddress where IsPrimaryShipFrom =1 and DoBusAsID = tblParameters.HostDoBusAsID
Host	HostStreet2	S	tblDoBusAsAddress.StreetAddress2 where IsPrimaryShipFrom =1 and DoBusAsID = tblParameters.HostDoBusAsID
Host	HostCity	S	tblDoBusAsAddress.City where IsPrimaryShipFrom =1 and DoBusAsID = tblParameters.HostDoBusAsID

Host	HostState	S	tblDoBusAsAddress.State where IsPrimaryShipFrom =1 and DoBusAsID = tblParameters.HostDoBusAsID
Host	HostZIP	S	tblDoBusAsAddress.ZipCode where IsPrimaryShipFrom =1 and DoBusAsID = tblParameters.HostDoBusAsID
Host	HostCountry	S	tblDoBusAsAddress.Country where IsPrimaryShipFrom =1 and DoBusAsID = tblParameters.HostDoBusAsID
Host	HostContact	S	tblDoBusAsAddress.ContactName where IsPrimaryShipFrom =1 and DoBusAsID = tblParameters.HostDoBusAsID
Host	HostPhone	S	tblDoBusAsAddress.Phone where IsPrimaryShipFrom =1 and DoBusAsID = tblParameters.HostDoBusAsID
Host	HostFAX	S	tblDoBusAsAddress.FAX where IsPrimaryShipFrom =1 and DoBusAsID = tblParameters.HostDoBusAsID
Host	UCC	U	Host UCC Code - Company Prefix for UPC labels and GTIN labels and SSCC labels
Host	CAGE	U	DoD Cage Code for Host
Host	UIDType	U	UID1 or UID2

### ***Attributes for Commercial Shipping Labels***

ContractNumber	from Contract
CustomerName	from Customer
ShipToName	from CustAddressName
ShipToStreet	from StreetAddress
ShipToStreet2	from StreetAddress2
ShipToCity	from City
ShipToState	from State
ShipToZIP	from ZipCode
ShipToCountry	from Country
ShipToContact	from ContactName
ShipToPhone	from Phone
DBACode	from DoBusAsCode
DBAName	from DoBusinessAsName
ShipFromName	Ship From Name
ShipFromStreet	Ship From Street
ShipFromStreet2	Ship From Street2

ShipFromCity	Ship From City
ShipFromState	Ship From State
ShipFromZIP	Ship From ZIP
ShipFromCountry	Ship From Country
ShipFromPhone	Ship From Phone
ShipFromFax	Ship From Fax
ShipFromContact	Ship From Contact
PayToName	Pay To Name
PayToStreet	Pay To Street
PayToStreet2	Pay To Street2
PayToCity	Pay To City
PayToState	Pay To State
PayToZIP	Pay To ZIP
PayToCountry	Pay To Country
PayToPhone	Pay To Phone
PayToFax	Pay To Fax
PayToContact	Pay To Contact
NetWeight	Weight of Container Contents
GrossWeight	Total Weight of Container
TareWeight	Weight of Empty Container
DateWanted	From Date Wanted field
Carrier	From Shipper
NumOfContainers	From Ship Order
ContainerSeqNo	Sequence Number of Container in Shipment
ContainerType	Description of Container - box, carton, pallet

If the container contains only a single part number then the following are also available:

PartNumber	Part Number of Contents of Container
PartDescription	Description of Contents if Single Use Container
PrimaryUnitCost	Unit Cost of Contents in Primary Unit of Measure
SecondaryUnitCost	Unit Cost of Contents in Secondary Unit of Measure
PrimaryQuantity	Quantity in Container in Primary Unit of Measure
SecondaryQuantity	Quantity in Container in Secondary Unit of Measure
PrimaryQuantityUOM	Unit of Measure for Primary Quantity, such as Pounds or Gallons
PrimaryQuantityUOMSF	Short Form for Unit of Measure for Primary Quantity, such as lbs or gals
SecondaryQuantityUOM	Unit of Measure for Secondary Quantity, such as Pounds or Gallons
SecondaryQuantityUOMSF	Short Form for Unit of Measure for Secondary Quantity, such as lbs or gals
ContainerBarcode	Tracking Barcode on Container
SerialNumber	Serial Number if individually barcoded item
LotNumber	Lot Number of Contents

All the above attributes can also appear as field names on a Military Shipment Label.

### Setting Up Military Shipping Labels

Some data attributes, such as the CAGE code attribute and the UCC/ECC/GS1 manufacturer identification code are setup as user attribute values at the Host or DBA level.

In order to generate iTCN and iPDF417, the value of the user defined attribute MSL must be set to “Yes”. All of the attributes needed for label fields must be present in the set of attributes, otherwise they will be requested at label generation time. The exception to this is the field names for the fields in the PDF417 barcode. These are all treated as optional and will not be inserted into the Barcode if not present.

The easiest way to setup the required contract attributes for printing an MSL on a contract is to create an Excel spreadsheet with the following data items. This can be imported into BellHawk using the Import Object Attribute Values function as described in the BellHawk Data Exchange (DEX) Users Manual.

Before importing the data, you will need to have created a contract in BellHawk with the specified contract number. Any field that you do not want present on the MSL should be omitted. Then save your data in a .CSV (comma delimited) format and import these attributes for the specified contract number.

	<b>Contract Number</b>	<b>Field Name</b>	<b>Value</b>
CONTRACT		TCNPrefix	
CONTRACT		Consignee1	
CONTRACT		Consignee2	
CONTRACT		Consignee3	
CONTRACT		Consignee4	
CONTRACT		Consignee5	
CONTRACT		TAC	
CONTRACT		Project	
CONTRACT		SHC	
CONTRACT		POE	
CONTRACT		POD	
CONTRACT		DoDAAC	
CONTRACT		TP	
CONTRACT		CDoDAAC	
CONTRACT		Mode	
CONTRACT		RDD	
CONTRACT		TCMDDIC	
CONTRACT		UNDIV	
CONTRACT		UNNAI	
CONTRACT		NATOID	
CONTRACT		CGC	
CONTRACT		TS	
CONTRACT		ACSHC	
CONTRACT		TPC	

The meaning of the field names and the format of the field values is specified in the companion document Military Shipment Label Field Names. As you will see there are many more field names than are listed above and you can add any of the Header Type 1, 6 or 7 codes to your above import.

Currently BellHawk does not generate line item data inside the 2D barcode and so the Comment field must be "NO LINE ITEM DATA" and Header type 8 fields are ignored. Field names are case sensitive.

Before the attributes values for an MSL can be imported they need to be defined. A file of definitions is being provided by BellHawk so the MSL definitions can be imported from an Excel spreadsheet.

Many of the fields in the MSL come from data set up for the customer contract/order within BellHawk. These are:

<b>Attribute Name</b>	<b>Default Value</b>	<b>From</b>
ShipTo1	ContactName	Customer Address
ShipTo2	CustomerName	Customer
ShipTo3	StreetAddress	Customer Address
ShipTo4	StreetAddress2	Customer Address
ShipTo5	City"+"State"+"ZipCode"+"Country	Customer Address
ShipToAddress	ShipTo1 "+"ShipTo2 "+"ShipTo3 "+"ShipTo4 "+"ShipTo5	Attributes
ShipDate	Today's date in dd-mon-yyyy format	dd is day in month, mon is JAN, FEB, MAR, etc and yyyy is year
ShipDateJ	Today's date in yddd format	Note that y is last digit of year and ddd is the day within the year
From1	DoBusAsName	Senders DBA
From2	StreetAddress	Senders DBA Address
From3	City"+"State"+"ZipCode"+"Country	Senders DBA Address
FromAddress	From1"+"From2"+"From3"	Senders DBA and Address
ConsigneeAddress	Consignee1"+"Consignee2" "+"Consignee3"+"Consignee4 "+"Consignee5"	User attributes for Container from Container Recordset
OriginPostalCode	ZipCode	Senders DBA Address
ShippingSequenceNo	ReleaseNumber	Ship Order
TotalPieces		From Ship Order
Piece	ContainerSeqNo"/"TotalPieces	From Container record

If a shipping container only holds a single part number then the following are also defined:

Nomenclature	UnitName	From Contract Line Item
LotNumber	LotNumber	From Container record
MSLContainerQuantity	Primary Quantity" "UnitNotation	From Contract Line Item
MSLPrice	UnitPrice"/"Unit Notation	From Contract Line Item

If the item being shipped is a serialized item then the following are defined:

PackageID	SerialNumber	From Container record
Nomenclature	UnitName	From Contract Line Item
LotNumber	LotNumber	From Container record
MSLPrice	UnitPrice"/"Unit Notation	From Contract Line Item

Default fields, such as the Ship To and From addresses can be over-ridden by defining these as user attributes on the Contract or Ship Order.

The weight of the container, the field GrossWeight, is computed by BellHawk from the tare weight of the container plus the weight of all items in the container.

The Cube of the container is usually specified as a user attribute Cube on the Container Type or is left to be requested at time of label generation.

### The Debug Capability

When label formats are first being set up, it is a good idea to turn on the debug flag.

**Edit Label Format**

Label Name :

Label Format File:  ...

Description:

Product  
 Product Container  
 General  
 Ship

Is Debug ?

Prior to a label being printed, this shows a report of the attributes that matched field names on the label and available field names that were not used:

**Label Attributes List**

*Label Format Name:* MSL                      *Label Format File Name:* \\Peters\laptop06\My Documents\BarTender\Formats\MSL.btw

*Date/Time:* 10/10/2006 12:57:46 PM                      *Tracking Barcode:* SW81238350D001XAE

<i>Attribute Name</i>	<i>Source Object Name</i>	<i>Attribute value</i>	<i>Is printed in Label</i>
<b>List of Attributes Printed/Used in Label</b>			
CAGE	DBA	0ZL12	Yes
CasePal	ContainerType	0	Yes
CDoDAAC	Contract	SW8123	Yes
Consignee1	Contract	SSgt John Smith	Yes
Consignee2	Contract	Spares Depot	Yes
Consignee3	Contract	Fort Monmouth	Yes
Consignee4	Contract	NJ 32884	Yes
Consignee5	Contract	USA	Yes
ContainerSeqNo	BellHawk	5	Yes
Cube	LabelField	3	Yes
DoDAAC	Contract	W55XGJ	Yes
GrossWeight	BellHawk	5	Yes
iPDF417	MSL	{}>06JKUSMSW81238350D001XAE3D62829KYellow Freight2LFred Smith+XYZ Distributing+123 Main Street+Anytown+ME+3126+USA3LBellHawk Systems Corporation+45 River Street+Millbury+MA+1527+USA5LSSgt John Smith+Spares Depot+Fort Monmouth+NJ 328851L16272Q513Q5/10725D0V26RM S27W55XGJ28228SW812330B3211934 TX135NO LINE ITEM DATA38Feet!	Yes
IRFID	LabelField		Yes

*Tuesday, October 10, 2006* *Page 1 of 7*

The above shows some of the attributes of a Military Shipping Label. Below are shown some of the available attributes that were not used.

**Label Attributes List**

*Label Format Name:* MSL                      *Label Format File Name:* \\Peters\laptop06\My Documents\BarTender\Formats\MSL.btw

*Date/Time:* 10/10/2006 12:57:46 PM                      *Tracking Barcode:* SW81238350D001XAE

TP	Contract	2	Yes
TypeService	Contract	TGBLUB	Yes
<i>Attribute Name</i>	<i>Source Object Name</i>	<i>Attribute value</i>	<i>Is printed in Label</i>
<b>List of Attributes NOT Printed/Used in Label</b>			
Carrier	ShipOrder	BB	No
Comment	MSL	NO LINE ITEM DATA	No
ConsigneeAddress	MSL	SSgt John Smith+Spares Depot+Fort Monmouth+NJ 3288	No
ContainerAge	Container	2	No
ContainerBarcode	Container	SW81238350D001XAC	No
ContainerType	Container	Roll	No

This is a great way of seeing what is available that could be incorporated into a label and for checking that the correct values are being printed on the label

For a Military Shipping Label, the debug report also details the contents of the PDF417 2D barcode:

<b><i>Label Attributes List</i></b>				
<b><i>Label Format Name:</i></b> MSL		<b><i>Label Format File Name:</i></b> \\Peters\laptop06\My Documents\BarTender\Formats\MSL.btw		
<b><i>Date/Time:</i></b> 10/10/2006 12:57:37 PM		<b><i>Tracking Barcode:</i></b>		
<b><i>MSL PDF 417 Fields</i></b>				
<i>Sequence No</i>	<i>Format</i>	<i>Data Indicator</i>	<i>Attribute Name</i>	<i>Value</i>
1	06	JKUSM	TCN	SW8 1238350D001XAE
2	06	3D	ShipDateJ	6282
3	06	9K	TAC	Yellow Freight
4	06	2L	ShipToAddress	Fred Smith+XYZ Distributing+123 Main Street+Anytown+ME+3126+USA
5	06	3L	FromAddress	BellHawk Systems Corporation+45 River Street+Millbury+MA+1527+USA
6	06	5L	ConsigneeAddress	SSgt John Smith+Spares Depot+Fort Monmouth+NJ 3268
7	06	51L	OriginPostalCode	1527
8	06	2Q	GrossWeight	5
9	06	13Q	Piece	5/1
10	07	25	POE	DOV
11	07	26	POD	RMS
12	07	27	DoDAAC	W55XGJ
13	07	28	TP	2
14	07	29	CDoDAAC	SW8123
15	07	30	Mode	B

Tuesday, October 10, 2006 Page 6 of 7

This is very useful in verifying that the correct fields are present in the label.

Once you are satisfied with the fields that are printed on the label then the debug flag can be turned off for that label for production printing.

## Supporting Materials

Example labels and attribute definition spreadsheets are available under the Supporting Materials link from the BellHawk website.

## Issues with BarTender

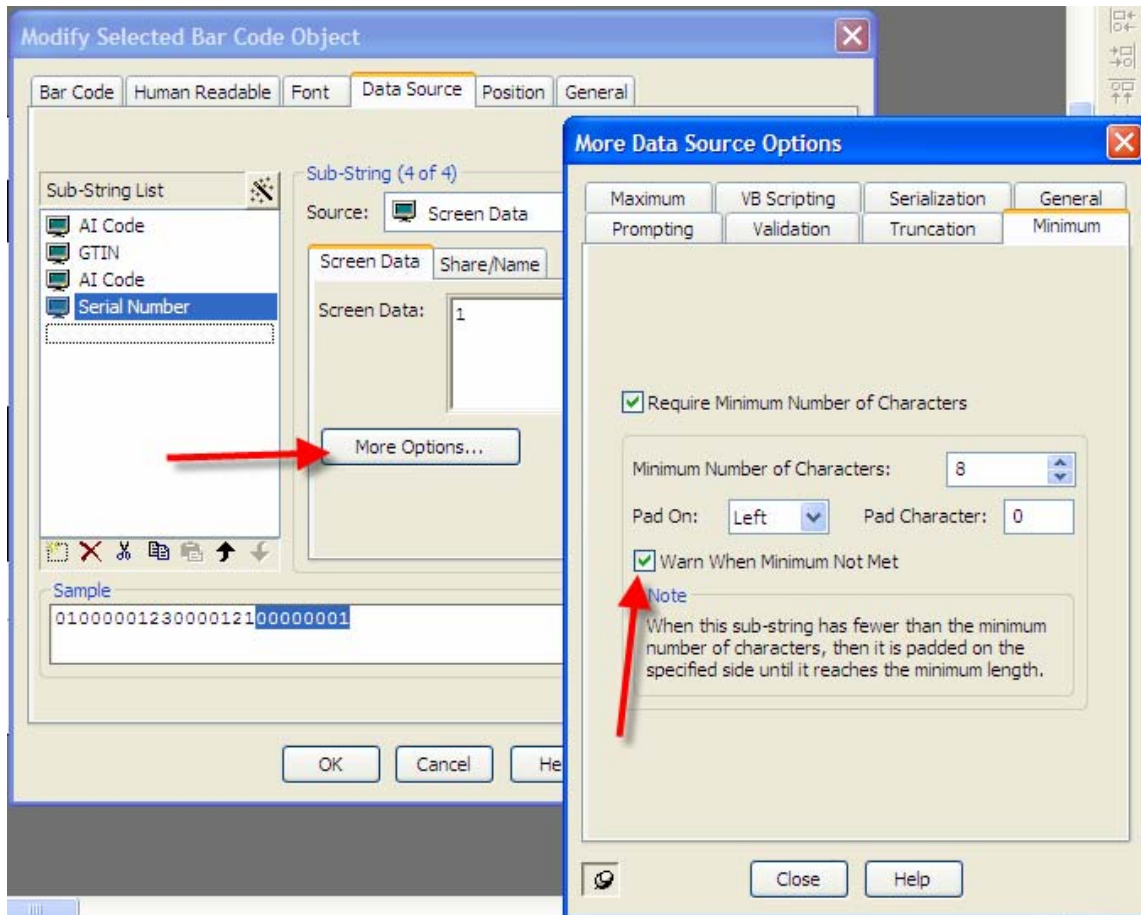
### ***Labels Not Printing from BellHawk***

BellHawk remotely calls BarTender using a call embedded in the VBA code on the PC. In doing this BarTender is run “Headless”, that is it is hidden from the users. An error output is written to the log file. Also, after printing an error message to the log-file (if one exists) BarTender hangs, waiting for an input in response to its error message.

Error messages are not relayed back to BellHawk when BarTender is called. Instead BarTender simply hangs waiting for input from the users (for which there is no indication).

The major cause of error messages are the Warning flags set on maximum and minimum and validation options of AI subfields in GIAI standard barcodes. These warning flags are set on by BarTender whenever you edit the subfields of a GIAI barcode.

To see these error flags, click on the More Options button in the field



And then select Minimum, Maximum, and Validation Tabs. You will find that BarTender has turned these Warnings on. Unfortunately what happens is that BarTender outputs these warnings to the BarTender log file and then hangs waiting for user input.

The way to correct this problem is to turn all the warning flags off in the Data Source Options. BarTender will then not print any warnings and hence the label will print without BarTender hanging.