



XML-Enabled 4.0 Reference Guide

Copyrights

Any unauthorized reproduction of the contents of this document, in part or whole, is strictly prohibited.

Limitation of Liability

SATO Corporation and its subsidiaries in Japan, the U.S. and other countries make no representations or warranties of any kind regarding this material, including, but not limited to, implied warranties of merchantability and fitness for a particular purpose. SATO Corporation shall not be held responsible for errors contained herein or any omissions from this material or for any damages, whether direct, indirect, incidental or consequential, in connection with the furnishing, distribution, performance or use of this material.

Specifications and contents in this document are subject to change without notice.

Trademarks

SATO is a registered trademark of SATO Holdings Corporation and its subsidiaries in Japan, the U.S. and other countries.

Oracle[®] is a registered trademark of Oracle Corporation.

SAP[®] is a registered trademark of SAP AG.

NiceLabel[®] is a registered trademark of Euro Plus d.o.o.

All other trademarks are the property of their respective owners.

Version: GBS-XMLEnabled4.0-r02-25-07-17RG

© 2017 SATO Corporation. All rights reserved.

Table of Contents

Table of Contents	1
Company Profile	3
Overview	4
1 Introduction	5
1.1 Introduction	5
1.1.1 About XML	5
1.1.2 Advantage with SATO XML-Enabled Printer	5
1.1.3 Printer Memory	5
1.2 Procedure	6
2 Printer Function - XML Data Stream	7
2.1 Document Type Definition.....	7
2.2 Samples of XML Data Stream	8
2.2.1 Oracle XML Data Stream	8
2.2.2 SAP XML Data Stream.....	9
2.3 Printer Setting	10
2.3.1 CL4NX/CL6NX Printer	10
3 Interface Solution.....	13
3.1 SATO Printing Solution for ERP Application	13
3.1.1 Solution 1: Middleware Solution with NiceLabel Automation Pro (LMS Pro)* ¹ Software	13
3.1.2 Solution 2: Direct Printing Approach with XML Parser	14
3.2 Choosing the Best Approach.....	15
4 NiceLabel Designer Pro.....	17
4.1 Overview	17
4.2 Using NiceLabel Designer Pro to Create Label Format.....	18
4.2.1 Create a New Label Design.....	18
4.3 Exporting Label Format to SATO XML-Enabled Printer	23
4.3.1 Setting the Format Number before Exporting to SATO XML-Enabled Printer.....	23
5 Glossary	29

This page is intentionally left blank.



Company Profile

SATO is a leading global provider of Auto-ID solutions that connect people, goods and information. It serves a diverse range of customers, delivering end-to-end solutions that streamline operations, empower workforces and help customers reduce their environmental impact.

For more information, please visit www.satoworldwide.com.

Overview

The SATO XML-enabled Printer Reference Guide contains the following sections:

- **Introduction**
The SATO XML-enabled printer is introduced in this chapter to provide a highlight of some of its features.
- **Printer Function - XML Data Stream**
In-depth information exploring the benefits of printing barcode or RFID smart labels with the SATO XML-enabled printers.
- **Interface Solution**
Implementation of the SATO XML-enabled printing for SAP, Oracle WMS and Mobile Supply Chain Application (MSCA) and other leading ERP applications.
- **NiceLabel Designer Pro**
XML label formats can be created using NiceLabel Designer Pro.
- **Glossary**
Provides a list of terms and definitions for your reference.

1

Introduction

The SATO XML-enabled printer is introduced in this chapter to provide a highlight of some of its features.

1.1 Introduction

1.1.1 About XML

Information in XML format provides greater flexibility as it can directly integrate with various systems. XML format offers flexibility as it is not confined to limited preset tags, which is contrasting to other markup languages with preset tags such as HTML. In other words, XML allows each vendor in a specified industry to create its own set of tags to cater to that industry without much sacrifice of compatibility issues. Comparing stored information is the traditional approach, where for example, a third-party software, typically the middleware, is required to store the data in an acceptable format into the database.

Now, large scale database systems, such as Oracle Warehouse Management System (WMS), can store information in XML format.

1.1.2 Advantage with SATO XML-Enabled Printer

Provide direct printing solutions with any database system without the need for any additional print server hardware and middleware application. The outcome is an overall printing system infrastructure that improves the efficiency of time and reduces unnecessary costs that would have otherwise been used to develop the middleware application.

Print labels from the Oracle Warehouse Management System (WMS) infrastructure or any ERP database application that can generate label information in XML format.

1.1.3 Printer Memory

The XML label format is stored in the memory that is residing in the printer. A standard SBPL feature is to download the label formats. The SBPL in the label format for the SATO XML-enabled printer is modified to include the necessary information that allows a link to be established between the XML variable data and the variable field on the label.

1.2 Procedure

- 1 A label is designed and created in XML format using the NiceLabel Designer Pro software.
- 2 The XML label format is exported to the SATO XML-enabled printer's memory by the means of the NiceLabel Designer Pro software.

- 3 The host sends a print job in XML format to the SATO XML-enabled printer over a network using an interface such as TCP/IP or by other means of connection.

The data sent from the host system to the SATO XML-enabled printer is analyzed by the printer's firmware to see if it is SBPL or XML data stream.

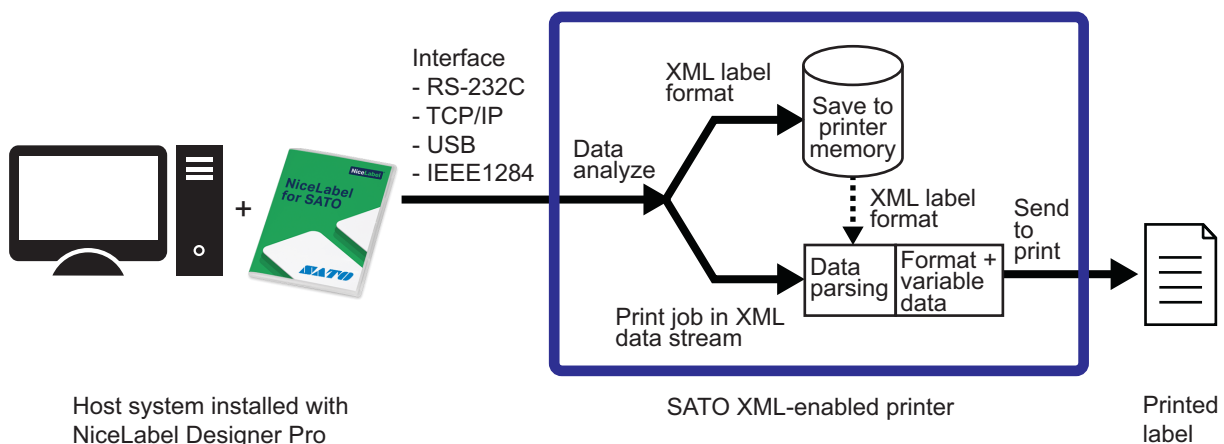
- 4 The SATO XML-enabled printer parses the XML print job data stream for the format attribute, label quantity, and variable field data.

When the XML data stream follows the Document Type Definition (DTD) semantics, it contains a <labels> or <label> element with a _FORMAT attribute.

- 5 The SATO XML-enabled printer will search for the label format number from its memory. The variable data field name is then inserted accordingly, where the format is recalled and print job begins (Merging the Format and XML variable data).

When the SATO XML-enabled printer has the requested format (stored in the memory), it recalls the format by interpreting the SBPL in the format. This will define the layout of the format (that is static text), and defines which fields are variables. The SATO XML-enabled printer then applies the variable field data from the XML data stream to the label format.

When end-of-label is detected in the XML data stream, the label is printed.



2

Printer Function - XML Data Stream

In-depth information exploring the benefits of printing barcode or RFID smart labels with the SATO XML-enabled printers.

2.1 Document Type Definition

XML DTD is outlined in this section, defining the implementation of an XML parser. The SATO XML-enabled printer will translate the XML according to this DTD. The purpose of a DTD is to define the legal building blocks of an XML document. DTD defines the document structure with a list of legal elements. A DTD can be declared in line in your XML document, or as an external reference.

The following briefly illustrates the purpose of XML.

- XML stands for **eXtensible Markup Language**
- XML is a **markup language** much similar to HTML.
- XML is designed to **describe data**.
- XML tags are not predefined in XML. You must **define your own tags**.
- XML uses a DTD (**Document Type Definition**) to describe the data.
- XML with a DTD is designed to be **self-describing**.

The printer name, label format, and duplicate quantities are specified in the <labels> tag initially to be used as a default value for each individual, <label> within the XML. Those default values can be overridden by each individual <label> tag for a specific printer name, label format, and duplicate quantities for that particular label print request. Note that unless values are specified within the <label> tag, the attributes (such as printer name, label format and quantity) will apply through the rest of the XML message.

```
<!ELEMENT labels (label)*>
<!ATTLIST labels _FORMAT CDATA #IMPLIED>
<!ATTLIST labels _JOBNAME CDATA #IMPLIED>
<!ATTLIST labels _QUANTITY CDATA #IMPLIED>
<!ATTLIST labels _PRINTERNAME CDATA #IMPLIED>

<!ELEMENT label (variable)*>
<!ATTLIST label _FORMAT CDATA #IMPLIED>
<!ATTLIST label _JOBNAME CDATA #IMPLIED>
<!ATTLIST label _QUANTITY CDATA #IMPLIED>
<!ATTLIST label _PRINTERNAME CDATA #IMPLIED>

<!ELEMENT variable (#PCDATA)>
<!ATTLIST variable name CDATA #IMPLIED>
```

2.2 Samples of XML Data Stream

2.2.1 Oracle XML Data Stream

The content of the XML message is in between <labels> and </labels>, which corresponds to a group of one or more label print requests. The content between <label> and </label> has the information to print one individual label. Each <variable> and </variable> pair gives the value of one variable on the label. The printer name, label format, and duplicate quantities are specified in the <labels> tag initially to be used as a default value for each individual, <label> within the XML. Those default values can be overridden by each individual <label> tag for a specific printer name, label format, and duplicate quantities for that particular label print request. Note that unless values are specified within the <label> tag, the attributes (such as printer name, label format and quantity) will apply through the rest of the XML message.

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE labels SYSTEM "label.dtd">
<labels _FORMAT="1" _QUANTITY="1" _PRINTERNAME="Printer1" _JOBNAME="LBL101">
  <label>
    <variable name= "ITEM">AS5488</variable>
    <variable name= "ITEMDESC">Desktop</variable>
    <variable name= "QTY">2</variable>
    <variable name= "UOM">EA</variable>
  </label>
  <label _FORMAT="LotLabel" _PRINTERNAME="Printer2" >
    <variable name= "ITEM">AS12345</variable>
    <variable name= "ITEMDESC">Mouse</variable>
    <variable name= "LOT">10200</variable>
    <variable name= "QTY">10</variable>
    <variable name= "UOM">Bx</variable>
  </label>
</labels>
```

As we can see from the above example, the XML data stream contains the <labels> element that defines the format, quantity, printer name, and job name. The SATO XML-enabled printer shall use the _FORMAT element, which only takes numeric digits, to recall the stored XML format from the memory, and the _QUANTITY element to determine the number of labels to be printed. Note that the LOT is one of several defined variable data fields, and the value, 10200, is the variable data to be printed on the label. The printer shall extract these variable data and match them with the predefined variables in the format that is recalled by the printer.

2.2.2 SAP XML Data Stream

The content of the SAP XML message is in between `<Command>` and `</Command>`, which corresponds to only one label print request. The content between `<FieldList>` and `</FieldList>` has the information to print one individual label. Each `<Field>` and `</Field>` pair gives the value of one field or variable on the label. The job name, label format, and print quantity are specified in the `<FieldList>` tag.

```
<?xml version="1.0" encoding="UTF-8"?>
<Command xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Command.xsd">
<WriteTagData readerID="Writer_Device">
  <Item>
    <FieldList format="C:Label.PL" jobName="Writer_Device20040929165746"
quantity="1">
      <Field name= "EPC">3074024220403B8000000008</Field>
      <Field name= "EPC_TYPE">SGTIN-96</Field>
      <Field name= "EPC_URN">urn:autoid:tag:sgtin-96:3.5.0037000.065774.8</
Field>
      <Field name= "PRODUCT">SGPROD</Field>
      <Field name= "PRODUCT_DESCRIPTION">Test product</Field>
    </FieldList>
  </Item>
</WriteTagData>
</Command>
```

As we can see from the above example, the XML data stream contains the `<FieldList>` element that defines the format, print quantity and job name. The XML-enabled printer shall use the "format" element to recall the stored XML format from the memory, and the "quantity" element to determine the number of labels to be printed. Note that "EPC" is one of several defined field or variable, and its content, 3074024220403B8000000008, is the field or variable data to be printed on the label. The printer shall extract these field or variable data and match them with the predefined fields or variables in the format that is recalled by the printer.

2.3 Printer Setting

The following procedure shows the printer setting for specified XML-enabled printer models.

2.3.1 CL4NX/CL6NX Printer

Before setting:

Check that the firmware version of the CL4NX/CL6NX XML-enabled printer is **1.6.1.r1** or later. (Refer to Information > Build Version in Chapter 4 Operation and Configuration of the CL4NX/CL6NX operator manual)

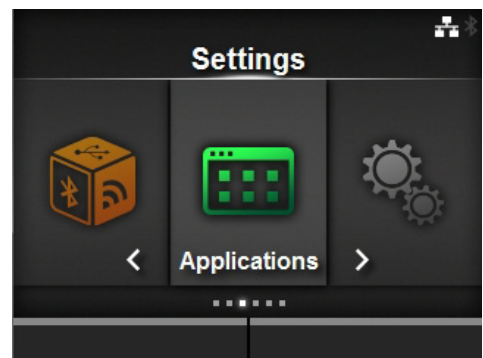
If the firmware version is of the older version, contact your SATO reseller or technical support center for details to upgrade to the latest firmware version.

1 When the printer is in online mode, press the **▶||** button to change the printer to offline mode.

2 Press the **←** button to show the **Settings** menu.

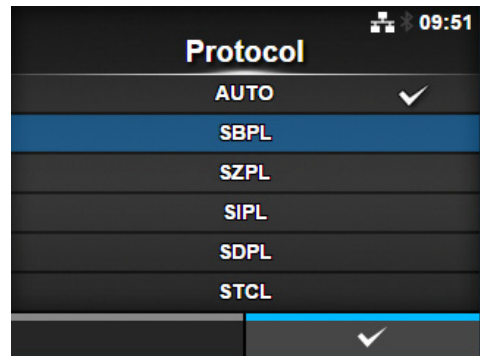
3 Press the **◀/▶** buttons to select **Applications** and then press the **←** button.

4 Press the **▲/▼** buttons to select **Protocol** and then press the **←** button.
The Protocol screen shows.



Applications	
Protocol	AUTO
SBPL	>
SZPL	>
SIPL	>
STCL	>
SDPL	>

- 5 Press the ▲/▼ buttons to select **SBPL**.
- 6 Press the right soft button or ← button to save the setting.
- 7 Press the ► button to return to offline mode. Press the ► button again to change the printer to online mode.



This page is intentionally left blank.

3

Interface Solution

Implementation of the SATO XML-enabled printing for SAP, Oracle WMS and Mobile Supply Chain Application (MSCA) and other leading ERP applications.

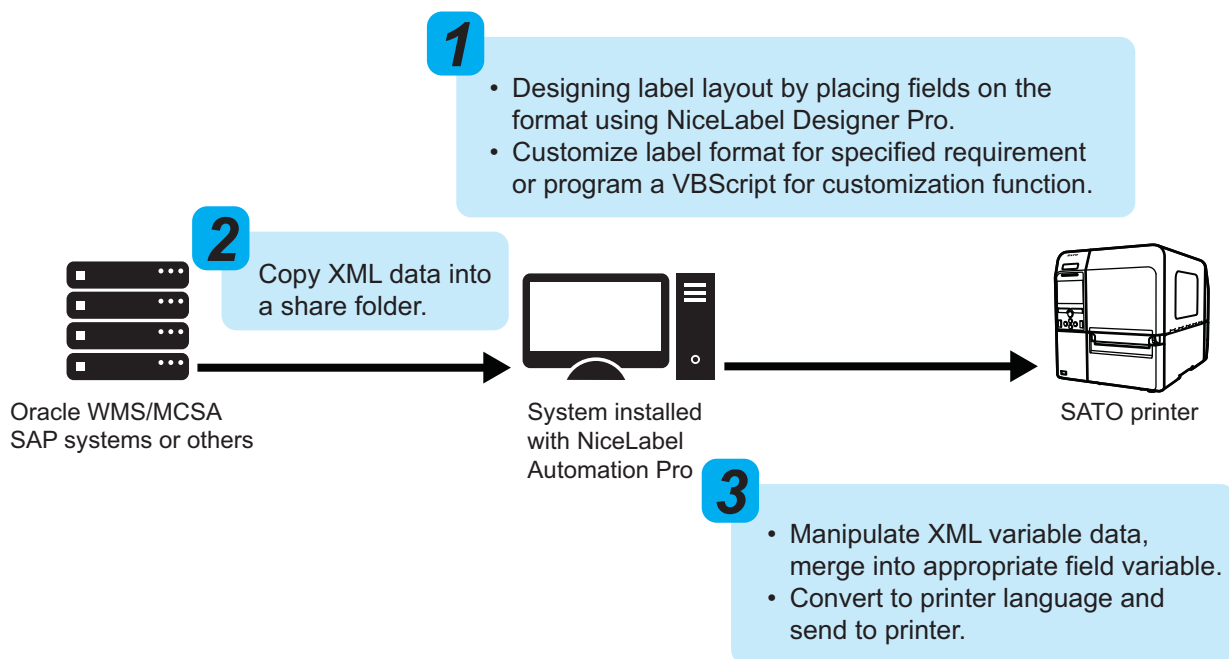
3.1 SATO Printing Solution for ERP Application

SATO offers two printing solutions interface to ERP application.

3.1.1 Solution 1: Middleware Solution with NiceLabel Automation Pro (LMS Pro)*¹ Software

Note

*¹ The NiceLabel Automation Pro is part of the module in the LMS (Label Management System) Pro software.



In this approach, NiceLabel Automation Pro (LMS Pro) acts a middleware solution to receive a XML data stream from ERP applications. NiceLabel Automation Pro (LMS Pro) is executed at the workstation. Each time a label printing is triggered in Oracle system or others, a XML file containing the label data is copied to a shared directory of workstation. NiceLabel Automation Pro (LMS Pro) will view the shared directory to check if any changes have been modified in the file. If any changes are detected, NiceLabel Automation Pro (LMS Pro) will extract the variable data from the XML file and merge it into the predefined label format. The printer driver will then convert it to the printer's language and transmit the data to the printer.

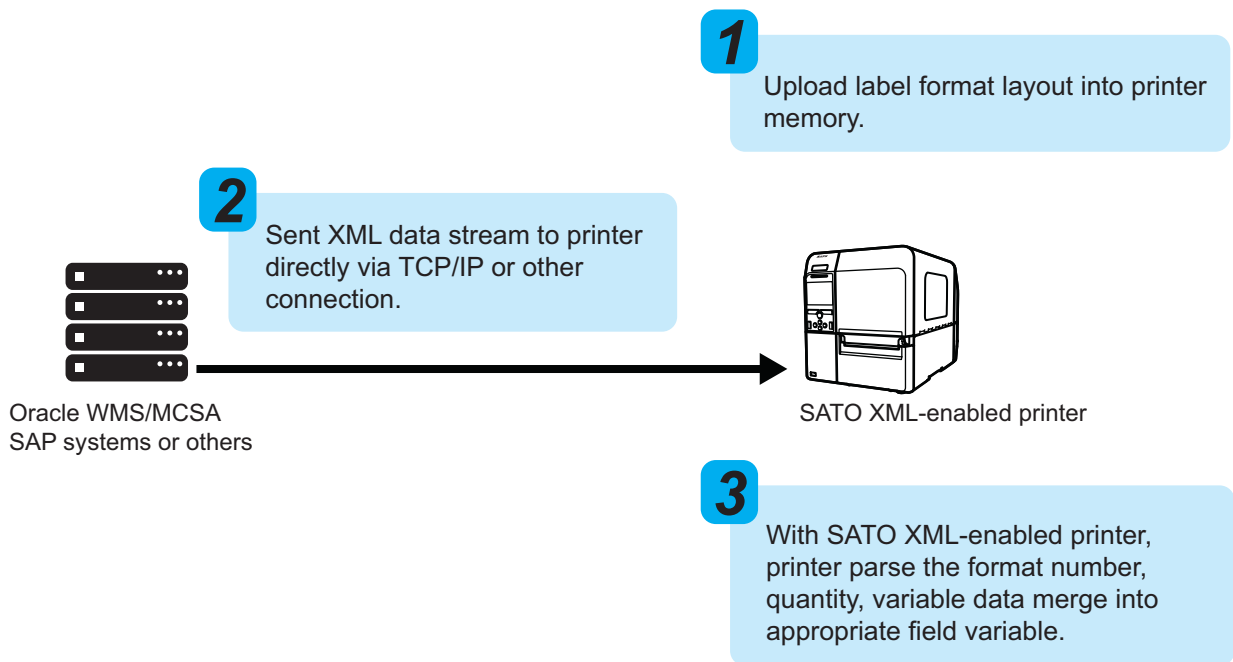
NiceLabel Designer Pro can be used to recreate a specified template format based on customized requests from different customers with differing needs. Hence, the customized programming can be done using this “acting” middle layer.

NiceLabel Automation Pro (LMS Pro) will manipulate the label at the appearance of trigger file from Oracle WMS/MSCA or others. This activity generates a trigger event for NiceLabel Automation Pro (LMS Pro) to start with action execution. NiceLabel Automation Pro (LMS Pro) provides a centralized label printing environment that easily integrates with business critical systems, without customization.

3.1.2 Solution 2: Direct Printing Approach with XML Parser

ERP applications use extensible markup language (XML) as the interface to exchange data with other applications. In order to connect SATO printer direct to such system such as Oracle WMS/MSCA and others without any additional print server or middleware, the SATO XML-enabled printer offers you a build-in XML parser firmware inside a printer. With XML parsing function, the SATO XML-enabled printer understands the native XML data stream generated by Oracle WMS/MSCA or any leading ERP system.

In this approach, the first step is to design the label format in NiceLabel Designer Pro with matching variable names. Next is to register a predefined format into the printer memory. Up to 999 formats can be registered, but this very much depends on the size of memory. Next, the label information in XML native format is then transferred to the SATO XML-enabled printer using TCP/IP, wireless or other connection. With XML parsing function, XML parser will analyzes the XML data stream that are the label format number, label quantity, and the variable data on the corresponding variable field. Then the SATO XML-enabled printer recalls the format and prints the barcode label.



3.2 Choosing the Best Approach

Solution 1: Middleware Solution with NiceLabel Automation Pro (LMS Pro) Software

Required items in this approach:

- Customize Java application that execute at Oracle WMS/MSCA or other system to give a triggering signal to the workstation.
- NiceLabel Designer Pro software to perform a format designing and NiceLabel Automation Pro (LMS Pro) software to execute printing.

Advantage

- Easy to customize the format layout.
- Filtering support for modifying the XML stream from Oracle WMS/MSCA and other leading ERP applications.
- Provide error notification to user if printing error occurs.

Solution 2: Direct Printing Approach with XML Parser

Required items in this approach:

- The SATO printer with XML parsing function enabled.

Advantage

- There are fewer layers, which simplifies the whole printing solution and minimizing costs.
- No additional middleware and print server is required.
- Deploy rapidly without customizing existing applications.

This page is intentionally left blank.

4

NiceLabel Designer Pro

XML label formats can be created using NiceLabel Designer Pro.

4.1 Overview

A purchased copy of NiceLabel Designer Pro can be used to create label formats with variables. Each designed label format can be exported to the SATO XML-enabled printer by classifying format number.

Important

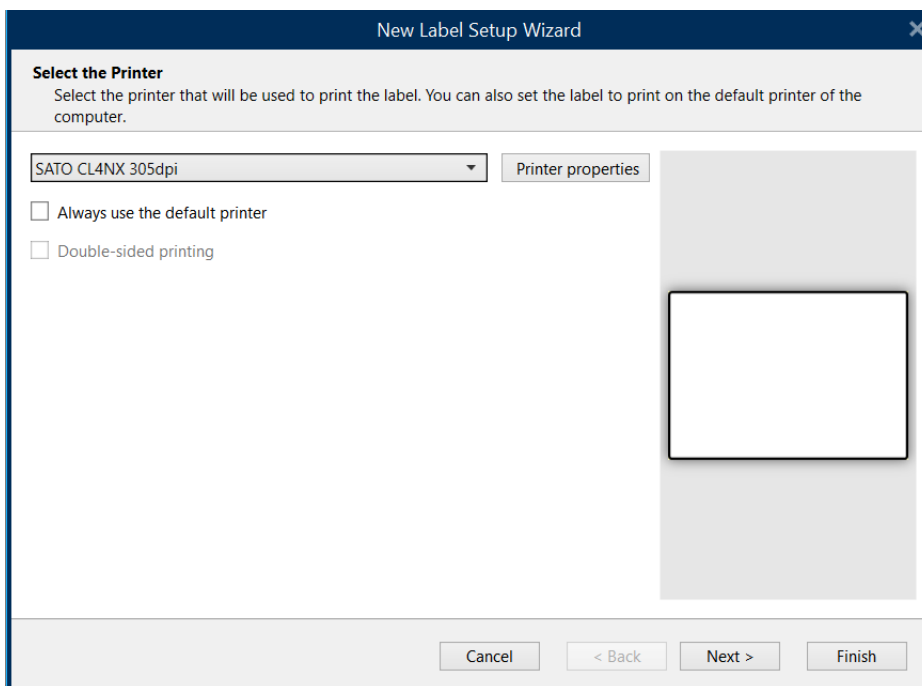
Before designing the label layout, you need to note the followings:

- Use only SATO internal fonts and barcodes.
- Design with field name that are exactly same as the XML file and they are case sensitive.

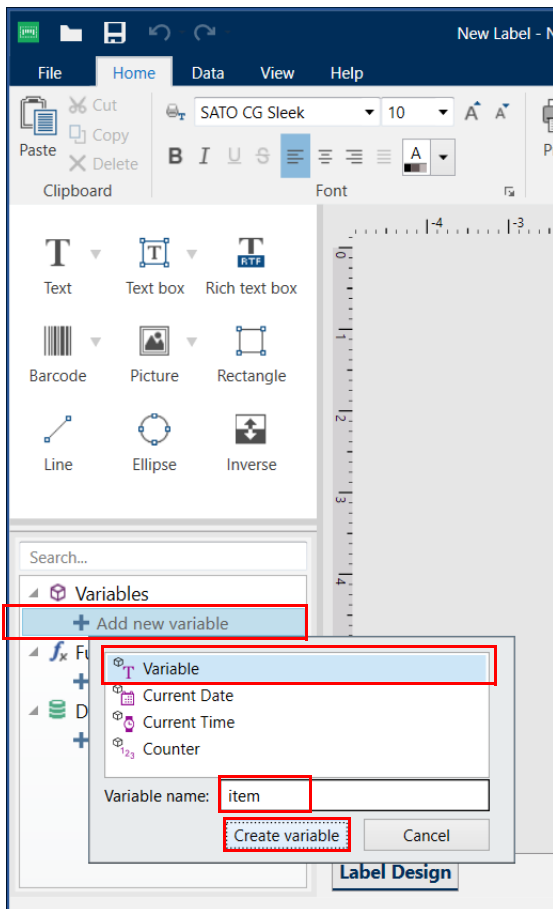
4.2 Using NiceLabel Designer Pro to Create Label Format

4.2.1 Create a New Label Design

- 1 Launch the NiceLabel Designer Pro and create new label.
- 2 Follow the instructions of the **Label Setup Wizard** to setup the target SATO XML-enabled printer, label layout and label dimensions.



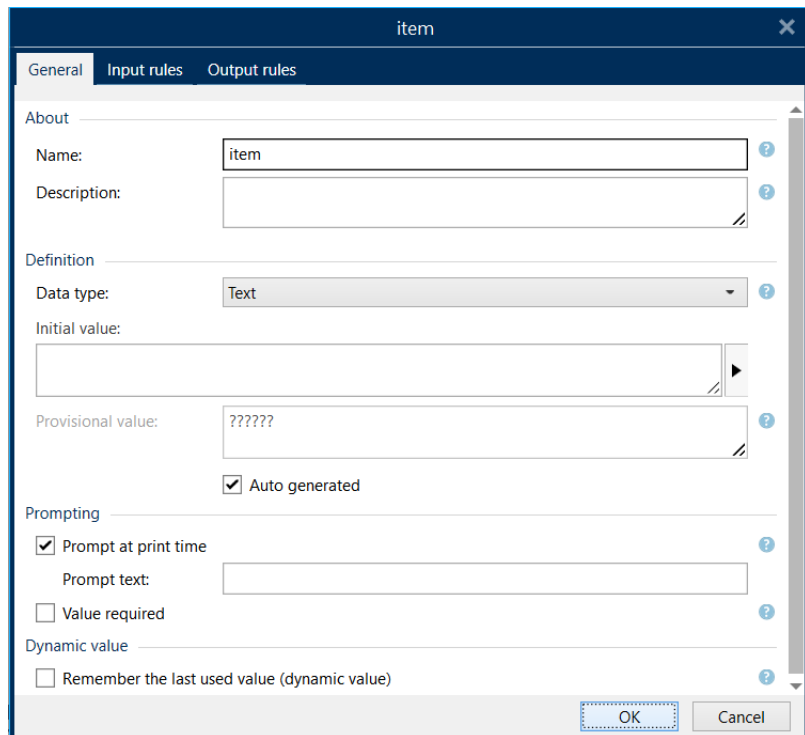
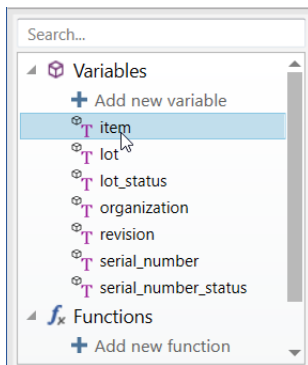
3 To add a XML variable data field, select **+ Add new variable** from the Data Explorer.



4 Select **Variable** and type the name of the variable according to the desired XML variable field in the XML file (See [“Fig. 2: Sample of the XML data stream for the serial label format.”](#) on page 28). Then click **Create variable**.

5 For creating additional XML variable, repeat step 3 and 4.

6 To edit variables, double-click the selected variable listed on the Data Explorer to open the properties window.

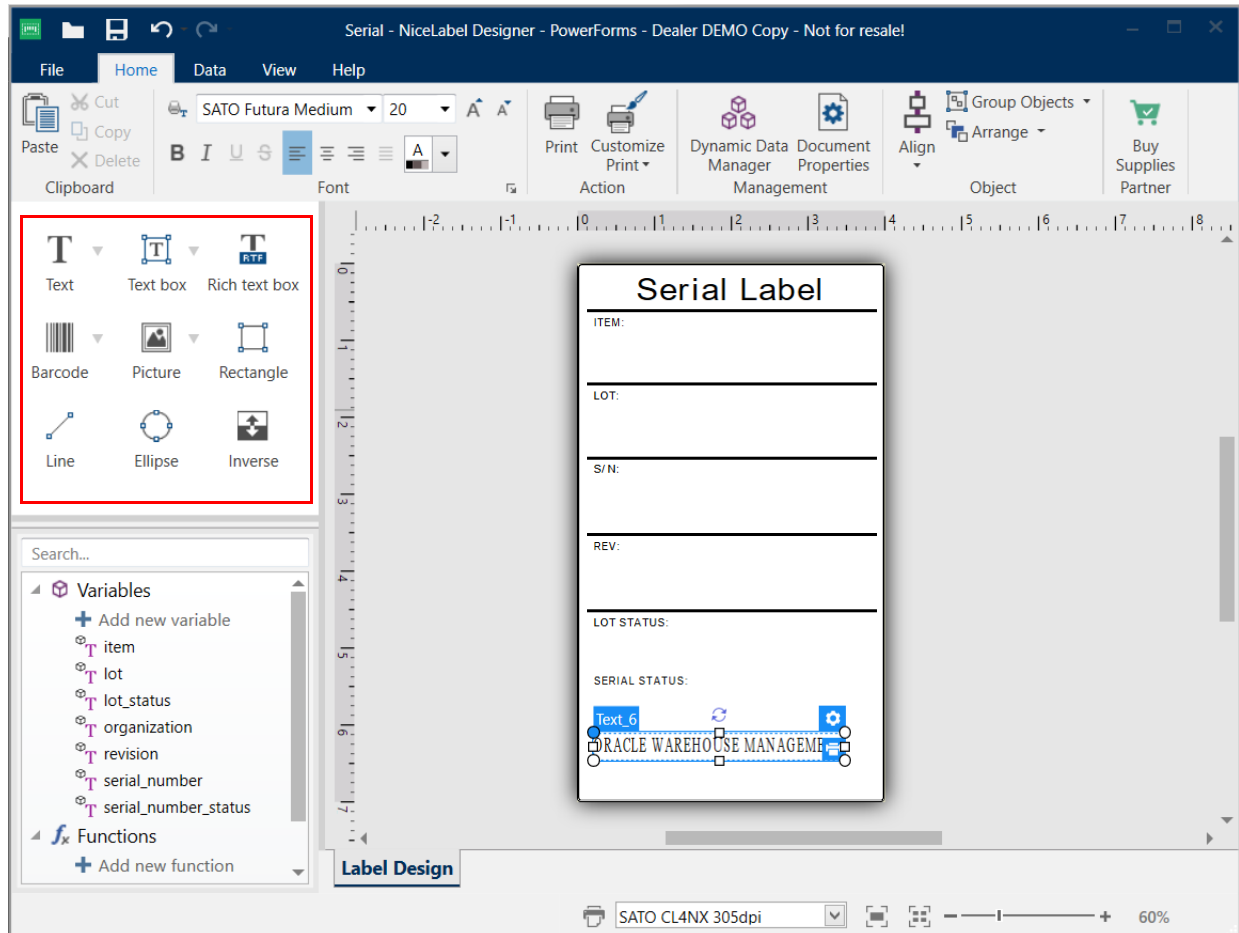


7 To design the rest of the label, first, create the static objects (text, graphics, barcode), which do not change with every label.

Select the object from the Object Toolbox and place it on the design surface.

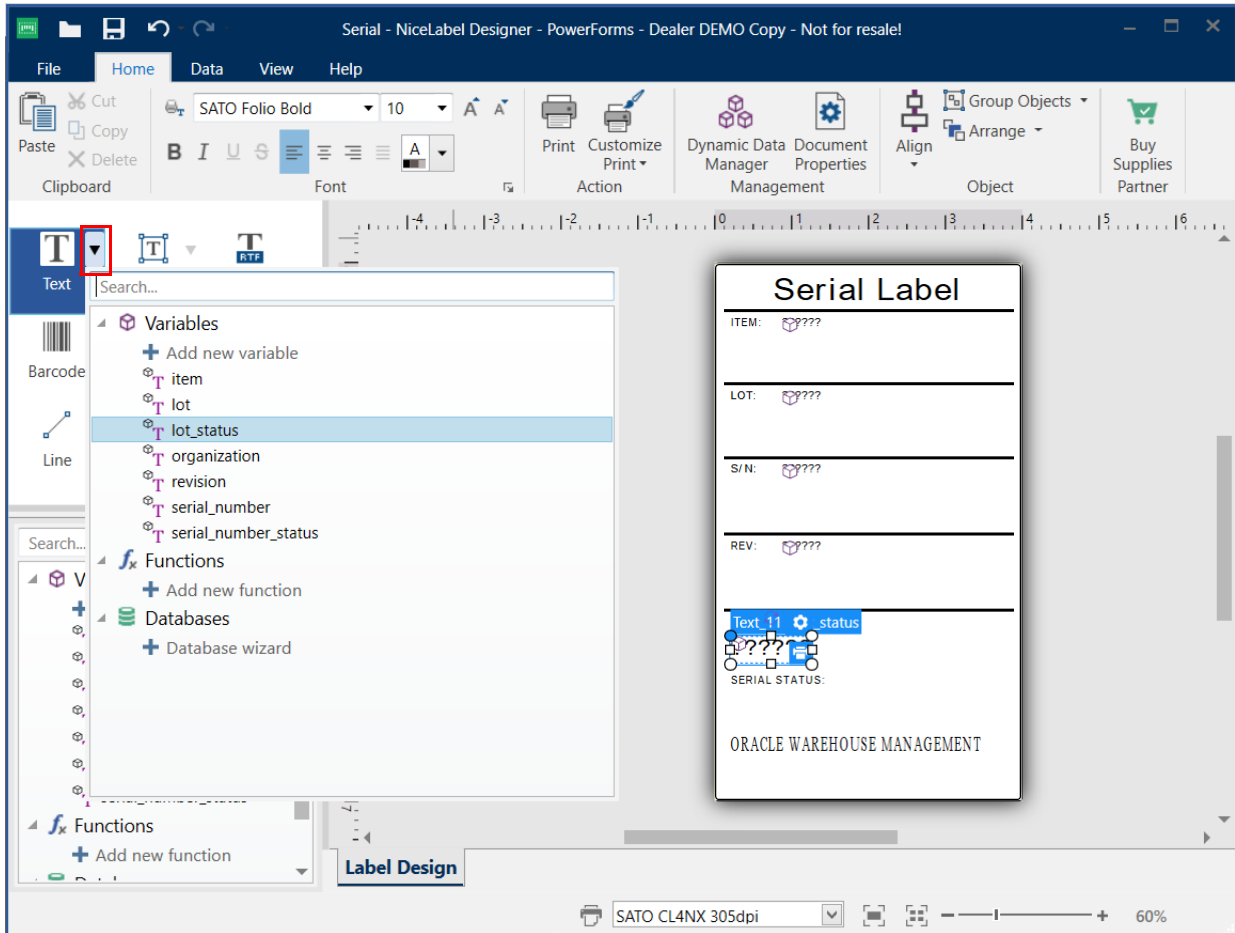
Refer to the NiceLabel Designer Pro user guide for more details.

NOTE: It is preferable to use SATO printer resident font for all the text objects.



8 To insert XML variable data fields, click the down arrow on the right side of the selected object icon and select the variable. Then place it on the design surface.

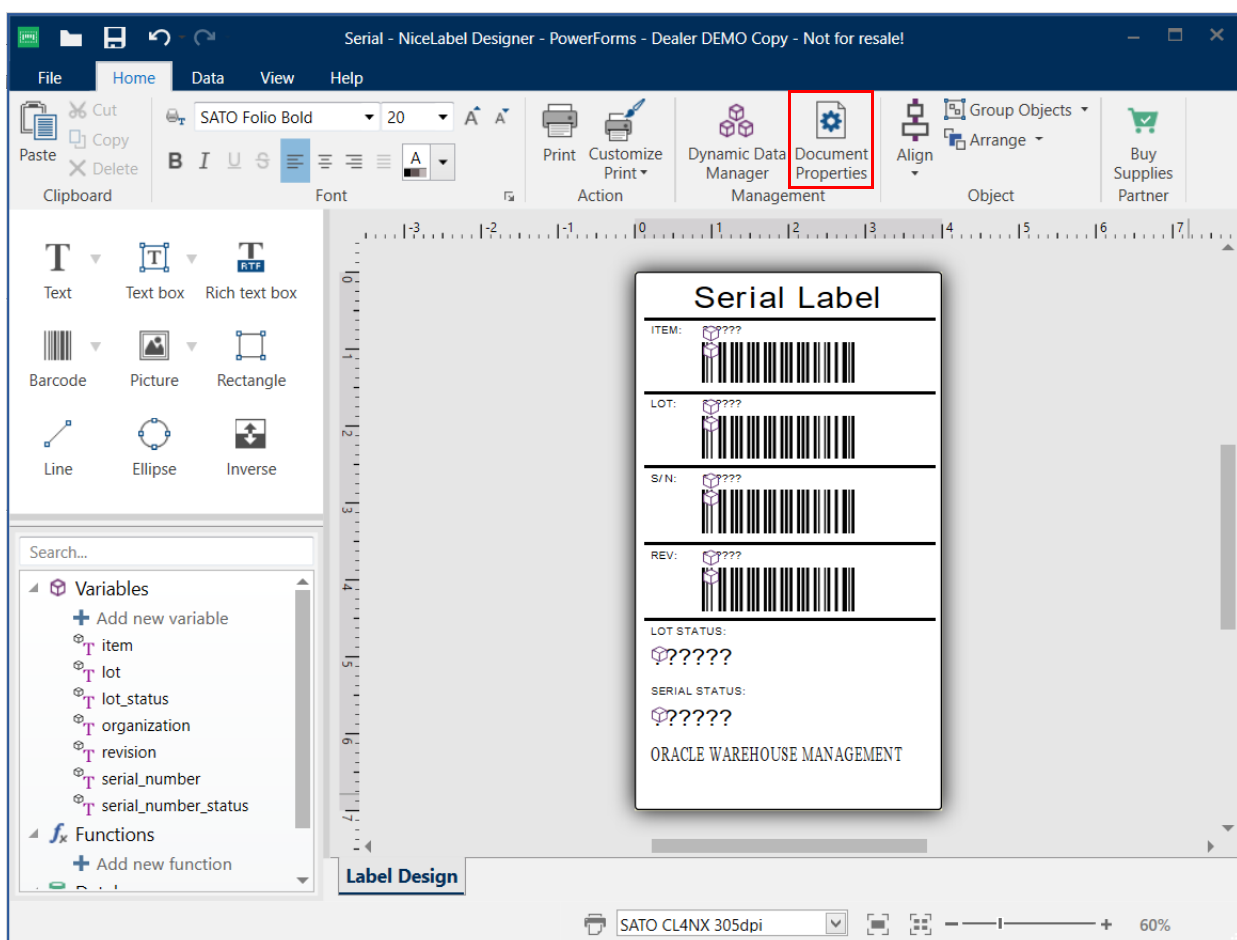
NOTE: Use only SATO printer resident font for all the text variables.



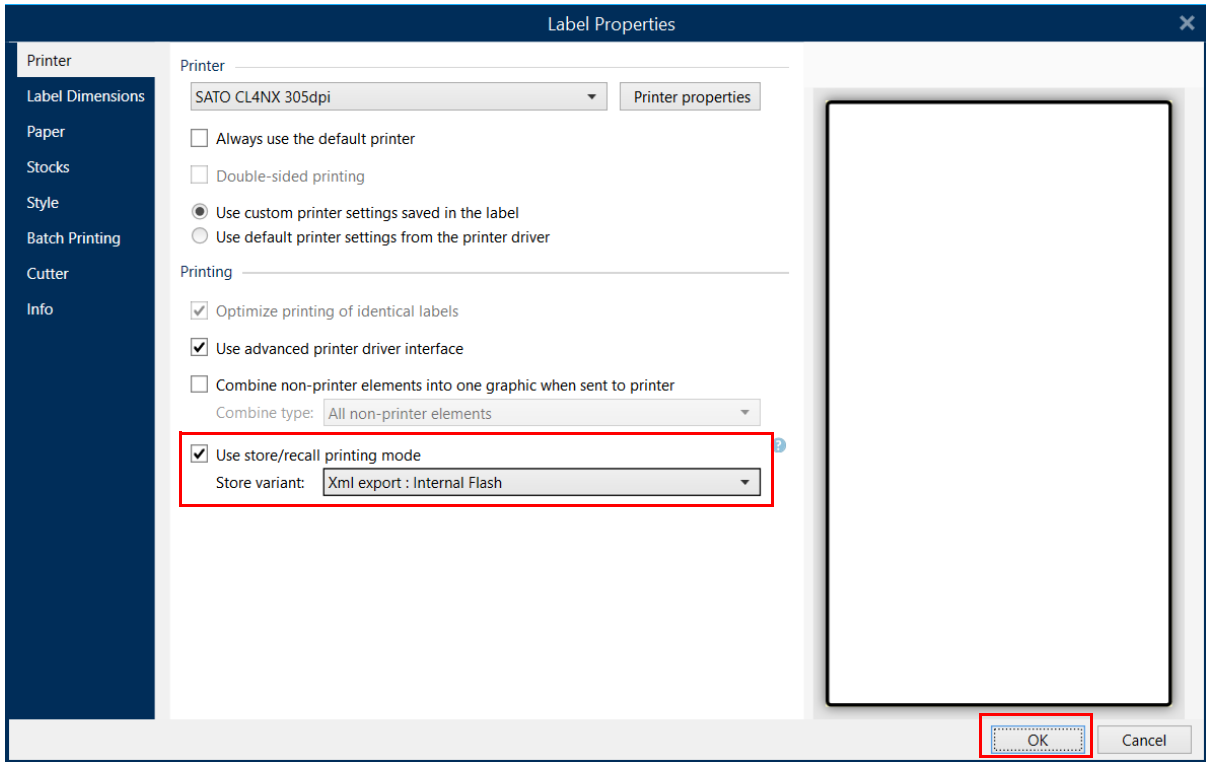
4.3 Exporting Label Format to SATO XML-Enabled Printer

4.3.1 Setting the Format Number before Exporting to SATO XML-Enabled Printer

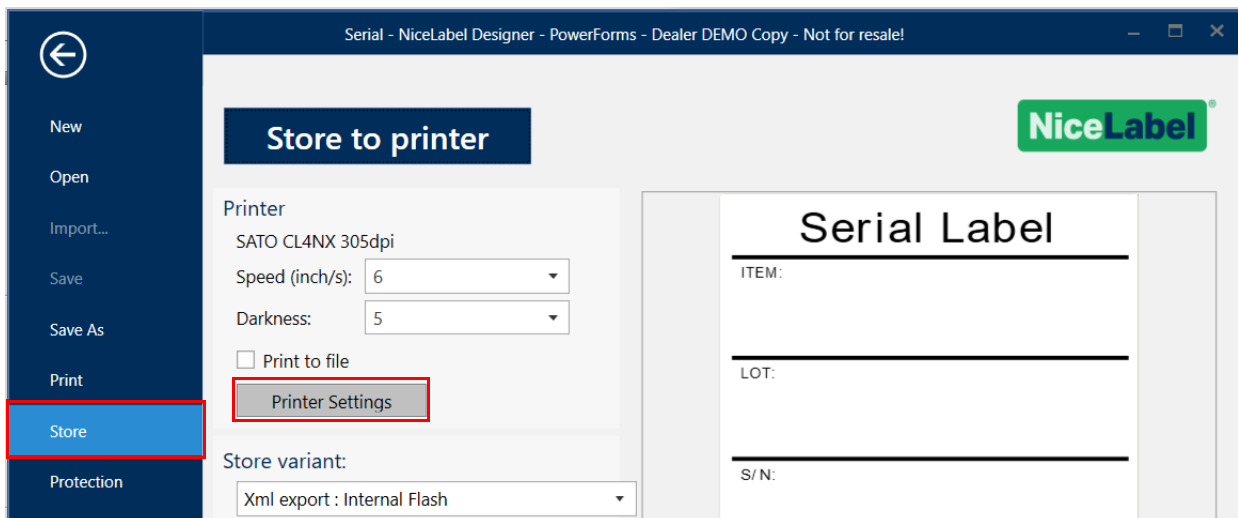
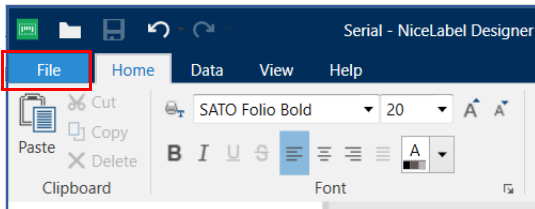
- 1 After the label format design is completed, click **Document Properties** to open the **Label Properties** window.



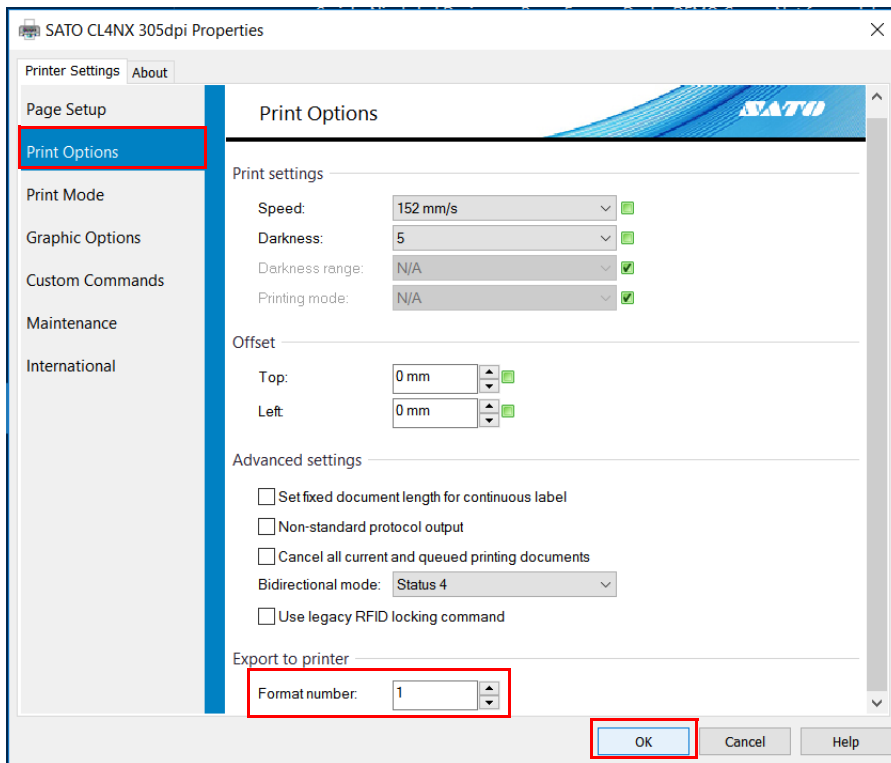
2 Select the **Use store/recall printing mode** check box and click **Xml export : Internal Flash** in the **Store variant** list. And then click **OK**.



3 Click **File** tab and select **Store** from the list. Then click **Printer Settings**.

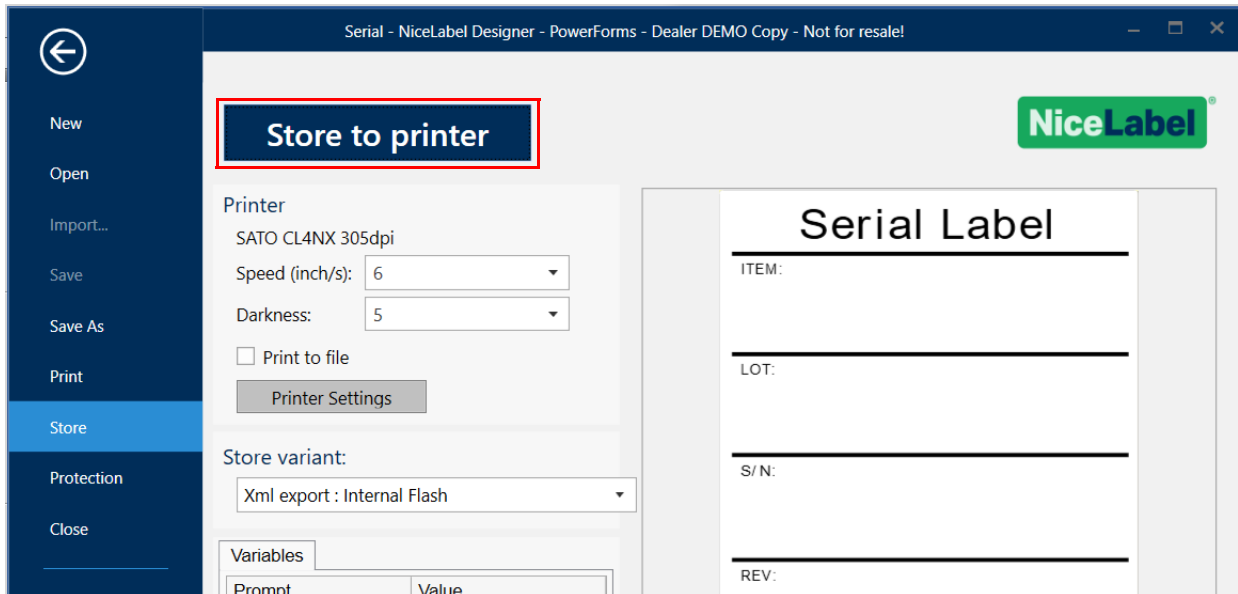


- 4 On the SATO printer **Properties** window, select **Print Options** from the **Printer Settings** tab. Set the **Format number** and click **OK**.

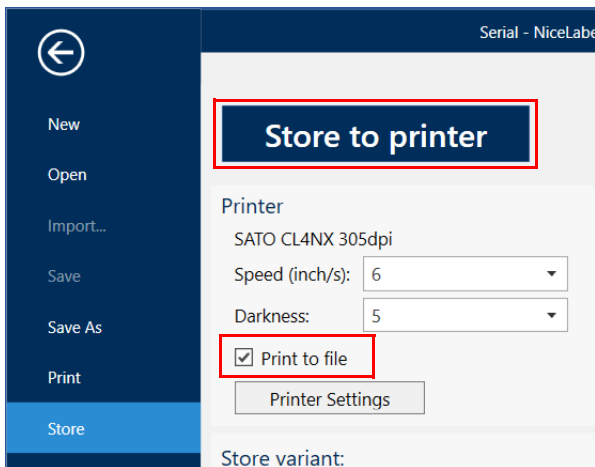


Use the same numeric digits (1 to 999) for labels_Format field in XML data stream.

5 Set the SATO XML-enabled printer to online mode. Click **Store to printer** to export the label format to the SATO XML-enabled printer.



If you select **Print to file** check-box and then click **Store to printer**, the label format is saved as PRN file.



```

h+A-CC2+*F,001+YS,001+Y,Serial-A1V02135H1220+A3V+000H+000+
%2+H0988+V02100+P02+RH0,SATOFOLIOB.ttf,0,109,120,Serial Label+%0-H0029+V01944+FW12H1154+
%2+H1158+V01928+P02+RH0,SATOFOLIOB.ttf,0,038,042,ITEM:+%0+H0029+V01652+FW12H1154+
%2+H1158+V01636+P02+RH0,SATOFOLIOB.ttf,0,038,042,LOT:+%0+H0029+V01359+FW12H1154+
%2+H1158+V01343+P02+RH0,SATOFOLIOB.ttf,0,038,042,S/N:+%0+H0029+V01054+FW12H1154+
%2+H1158+V01038+P02+RH0,SATOFOLIOB.ttf,0,038,042,REV:+%0+H0029+V00752+FW12H1154+
%2+H1158+V00735+P02+RH0,SATOFOLIOB.ttf,0,038,042,LOT STATUS:+
%2+H1158+V00500+P02+RH0,SATOFOLIOB.ttf,0,038,042,SERIAL STATUS:+
%2+H1158+V00273+P02+RH0,SATOFUTURAMC.ttf,0,081,085,ORACLE WAREHOUSE MANAGEMENT+X,01,06,item+
%2+H0954+V01928+P02+RH0,SATOFOLIOB.ttf,0,038,042,+X,02,06,lot+
%2+H0954+V01636+P02+RH0,SATOFOLIOB.ttf,0,038,042,+X,03,06,serial_number+
%2+H0954+V01343+P02+RH0,SATOFOLIOB.ttf,0,038,042,+X,04,06,revision+
%2+H0954+V01038+P02+RH0,SATOFOLIOB.ttf,0,038,042,+X,05,06,lot_status+
%2+H1158+V00656+P02+RH0,SATOFOLIOB.ttf,0,078,085,+ZL
    
```

Fig. 1: Sample of label format (in SBPL codes) exported to SATO XML-enabled printer.

6 Select **Print** from the list and then enter the value of all the variables.

The screenshot shows the NiceLabel Designer Pro interface. On the left is a dark blue sidebar with a menu containing: New, Open, Import..., Save, Save As, **Print** (highlighted with a red box), Store, Protection, Close, Options, About, and Exit. The main window title is "Serial - NiceLabel Designer - PowerForms - Dealer DEMO Copy - Not for resale!". The top right corner features the SATO logo and "SATO GLOBAL".

The central area is the "Print" dialog. It includes a printer selection dropdown set to "SATO CL4NX 305dpi", "Speed (inch/s): 6", and "Darkness: 5". There is a checkbox for "Print to file" and a "Printer Settings" button. Below these are quantity options: "Number of labels" (set to 1), "Number of pages", and "Print all labels (unlimited)". A "more..." link is also present.

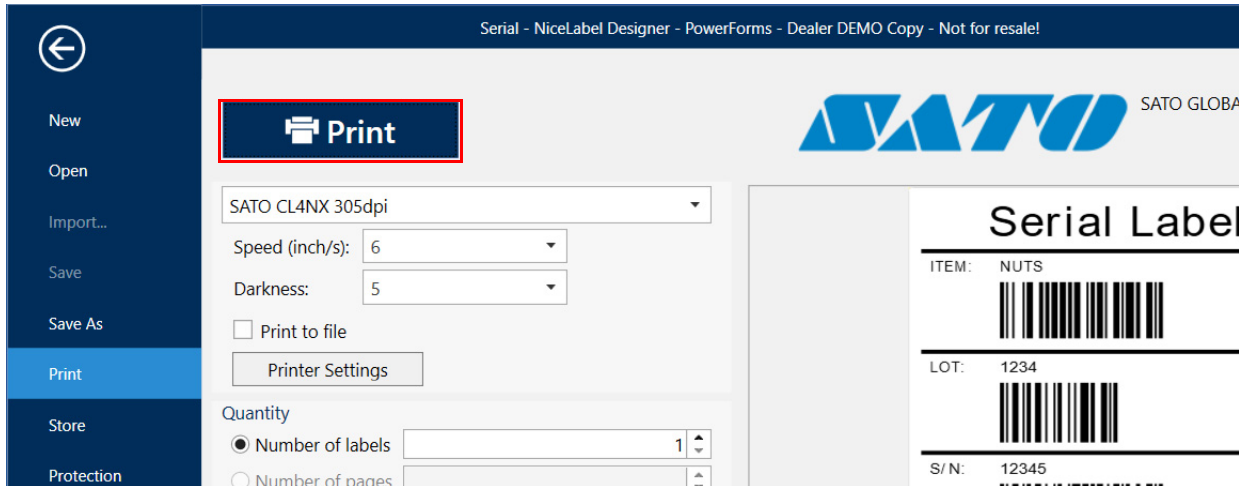
At the bottom of the dialog is a "Variables" table:

Prompt	Value
item	NUTS
lot	1234
lot_status	1234
revision	V2
serial_number	12345
serial_number_status	Active

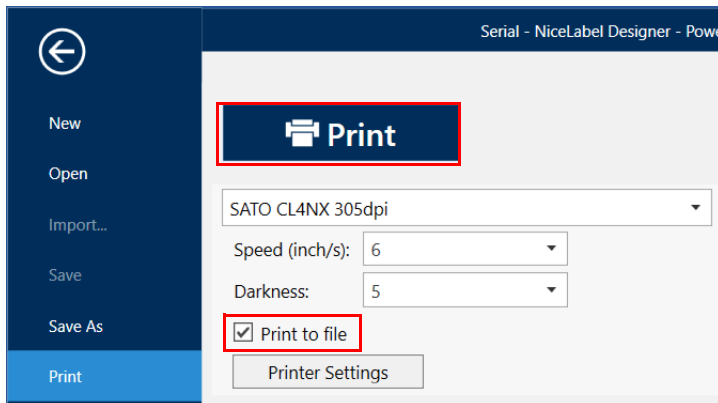
On the right side of the dialog, a preview of a "Serial Label" is shown. It contains the following information:

- ITEM: NUTS (with a barcode)
- LOT: 1234 (with a barcode)
- S/N: 12345 (with a barcode)
- REV: V2 (with a barcode)
- LOT STATUS: 1234
- SERIAL STATUS: Active
- ORACLE WAREHOUSE MANAGEMI

7 Then click **Print** to print out the label with entered variables.



If you select **Print to file** check-box and then click **Print**, the label is saved as XML script data. This data can be used as a sample reference of XML data stream for system developer to configure ERP system or Oracle for label printing.



```
M:\XML_Enabled_4.0_References\Serial.prn - Notepad++
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
Serial.prn x
1 <?xml version="1.0" standalone="no"?>
2 <!DOCTYPE labels SYSTEM "label.dtd">
3 <labels _FORMAT="001" _QUANTITY="1" _PRINTERNAME="Printer 1" _JOBNAME="Serial">
4   <label>
5     <variable name="item">NUTS</variable>
6     <variable name="lot">1234</variable>
7     <variable name="serial_number">12345</variable>
8     <variable name="revision">V2</variable>
9     <variable name="lot_status">1234</variable>
10    <variable name="serial_number_status">Active</variable>
11  </label>
12 </labels>
13
eXtensi length : 513 lines : 13 Ln: 1 Col: 1 Sel: 0 | 0 Windows (CR LF) UTF-8 INS
```

Fig. 2: Sample of the XML data stream for the serial label format.

5

Glossary

Provides a list of terms and definitions for your reference.

Data stream	A continuous flow of information or data transmitted on a byte-to-byte basis.
DDE (Dynamic Data Exchange)	A protocol that enables communication between Windows applications. An application must support DDE in order to use it. With DDE Commands, you can control one application (for example NiceLabel Designer Pro) with another (for example Microsoft Access).
Document Type Definition (DTD)	States what tags and attributes are used to describe content in an XML document, where each tag is allowed, and which tags can appear within other tags.
Firmware	Computer programming instructions that is stored in a read-only memory unit rather than being implemented through software.
JOB file	This is a plain ASCII text file containing NiceLabel commands, which are to be executed. JOB files are usually used to print a batch of labels.
Middle Layer (Middleware)	Software agents acting as an intermediary between different application components. It is used most often to support complex, distributed applications.
NiceLabel Automation Pro (LMS Pro)	The NiceLabel Automation Pro is part of the module in the LMS (Label Management System) Pro software. NiceLabel Automation Pro (LMS Pro) acts as middleware solution to receive a XML data stream from ERP applications. NiceLabel Automation Pro (LMS Pro) will manipulate the label at the appearance of trigger file.
NiceLabel Designer Pro	A barcode design and printing software that lets you create and print labels and fully supports WYSIWYG (What You See Is What You Get). Using NiceLabel Designer Pro, you can easily design any label to include bar codes, text, lines, boxes and graphics.
Parse	To break down into smaller component parts with an explanation of the form, function, and syntactical relationship of each part.
RS-232C	An interface used for serial communications such as between the printer and the computer for light-duty applications. It is also known as "serial cable".
SBPL	SATO native command language that allows users to produce high-quality labels using its printer control language and label definitions.

- TCP/IP** A protocol for communication between computers, used as a standard for transmitting data over networks and as the basis for standard Internet protocols.
- Trigger event** Event that must occur before NiceLabel Automation Pro (LMS Pro) begins executing some actions - for example printing labels. There are five types of events: file, serial port, database, TCP/IP server and HTTP server events.
- File trigger event occurs when a specified trigger file is modified or created. Serial port trigger event occurs when data is received on the monitored RS232 serial port. Database trigger event occurs when a change in the monitored database table is detected. TCP/IP and HTTP server trigger event occurs when data is received on the monitored socket (IP address and port number).
- Trigger file** NiceLabel Automation Pro (LMS Pro) monitors the modification time and date of the trigger file and if the time of the trigger files is newer than the time the action is last executed, it executes the action. Changing the time and date of the file is called "touching the file". One of the ways to perform that is using the MS-DOS command: REM>trigger_file.
- XML** A metalanguage with the acronym, eXtensible Markup Language, that allows one to design a markup language, used to allow for the easy interchange of documents on the World Wide Web.

This page is intentionally left blank.



Extensive contact information for worldwide
SATO operations can be found on the Internet at
www.satoworldwide.com

SATO