



QAD Adaptive Applications

User Guide

QAD Automation Solutions: Label Printing Services

Overview
Label Print Setup
Processing with Label Printing
Using Labels as Templates

This document contains proprietary information that is protected by copyright and other intellectual property laws. No part of this document may be reproduced, translated, or modified without the prior written consent of QAD Inc. The information contained in this document is subject to change without notice.

QAD Inc. provides this material as is and makes no warranty of any kind, expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. QAD Inc. shall not be liable for errors contained herein or for incidental or consequential damages (including lost profits) in connection with the furnishing, performance, or use of this material whether based on warranty, contract, or other legal theory.

This document contains trademarks owned by QAD Inc. and other companies.

Copyright ©2021 by QAD Inc.

LabelPrintingServices_UG_v033.pdf/crl/crl

QAD Inc.

100 Innovation Place
Santa Barbara, California 93108
Phone (805) 566-6000
<https://www.qad.com>

Contents

Label Printing Services User Guide Change Summary	vii
Chapter 1 Overview	1
Overview	2
Business Requirements/Needs	2
Solution	2
Features	5
Barcode Labeling Service	6
Out-of-the-Box Labels as Templates	6
Third-Party Templates	6
Configure Label Printing Services	7
Configure Syntax Types and Format	7
Monitor and Track Label Printing Data	8
System Tokens	8
Label Printing Programs	9
Limitations	10
Chapter 2 Label Print Setup	11
Setup Overview	12
Set Up Label Printing Master Data	13
Connect to an AppServer	13
Set Up Printers	14
Label Printer Connection Maintenance	14
Label Printer Syntax Type Maintenance	14
Printer Model Maintenance	15
Print Queue Maintenance	15
Label Printer Maintenance	16
Define Label Formats	18
Label Template Maintenance	19
Label Format Maintenance	21
Delete Label History Records	25
Label Dataset Import	26
Label Dataset Maintenance	26
Label Mapping Functions	27
Label Mapping	28

Configure Applications	29
Set Up Contextual Routing	30
Label Content Routing Setup	31
Label Content Routing Configuration	33
Set Up Contextual Printer Routing	35
Label Printer Routing Setup	36
Label Printer Routing Configuration	38
Set Up Label Extraction	40
Label Type Maintenance	40
Label Group Maintenance	41
Label Event Maintenance	42
Label Event Processing	43
Label Event Type Criteria Maintenance	43
Extraction Process	45
API and Processing	45
Make Calls without Considering PROPATH	45
Import/Export Label Setup Data	46
Label Config Import/Export	46
Label Extraction Import/Export	47
Delete Stranded Work Flows	48
Set Up Labels for QAD Serialization	48
Label Event Workfile	49
Data Collection Transactions	49

Chapter 3 Processing with Label Printing51

Processing Overview	52
View Label Print Status and Reprint Labels	53
Label Print Status Collection	53
View Labels	56
Label Request History	56
Label Queue Monitor	57
Label Maintenance	58
Label Reprint Maintenance	59
View Router Configuration Data	59
Print Labels Manually	60
Label Manual Release	60
Manual Label Reprint	62
Ad Hoc Label Generation	63
Automatically Start and Release Requests from Queues	64
Making Calls to the Label Printing Services	65

Chapter 4 Using Labels as Templates	67
Label Overview	68
Label Specifications	68
Label Descriptions	68
Import Label Data	69
Import Labels	69
Import Printer	71
Import Format	72
Import Extraction Configuration	73
Configuration	74
Extraction Programs	75
Overview	75
Include Files	75
Input Parameters	76
Provided Extraction Programs	76
Labels	78
Customer Item Label	79
Purchase Order Label	79
Serial Label	80
Rack Label	81
Blend Label	82
SO Labels	83
Master Label	83
Mixed Label	85
Container Label	85
Destination Label	87
Serial Detail Label	88
Truck Label	89
DRP Labels	90
DO Container Label	90
DO Destination Label	91
DO Master Label	92
DO Master Serial Detail	93
DO Mixed Serial Label	95
DO Truck Label	96
Index.....	99



Label Printing Services User Guide Change Summary

Product Name Changes

Starting in September 2019, the new name for QAD's complete portfolio of products is QAD Adaptive Applications. Additionally, QAD Adaptive ERP is the new name for QAD's flagship ERP solution. QAD Adaptive ERP includes the functionality previously associated with QAD Cloud ERP and QAD Enterprise Applications - Enterprise Edition, plus the QAD Enterprise Platform and Adaptive UX which resulted from the Channel Islands program. Going forward, the terms QAD Enterprise Applications, QAD Cloud ERP, and Channel Islands will be deprecated but will remain in previous documentation and training materials. QAD's intention is to—as soon as possible—eliminate the use of the deprecated terms going forward.

Change Summary

The following table summarizes significant differences between this document and previous versions.

Date/Version	Description	Reference
December 2020/Version 3.3-Revision 2	Updated Extraction program information in Chapter 4	page 75
	Added additional information for making calls to Label Printing Services.	page 45 page 65
October 2020/Version 3.3-Revision 1	Updated list of input parameters in Extraction Programs section.	page 76
October 2020/Version 3.3	Rebranded book for version 3.3.	--
	Removed references to EE 2015; version 3.3 supports EE 2106 and higher.	--
	Updated text for the URL field and the Script field in Label Printer Maintenance.	page 16
	Added new setup processing and configuration information for serial labels.	page 48
	Added information for configuring out-of-the-box labels.	page 74
	Added a new Extraction Program section to Chapter 4 to describe the purpose, dataset created, tables queried, input parameters, dataset fields, and more.	page 75
	Added new distributed requirements processing (DRP) distributed order (DO) labels.	page 90
	Added new item number and quantity to the Serial Label template.	page 80

Date/Version	Description	Reference
	Added two new Rack and Blend labels as SO labels to use as templates.	page 81 page 82
September 2019/version 3.2	Added text for a new Remove Stranded Label Work File Records utility.	page 48
	Added text for a new copy function in the Action drop-down menu for Label Content Routing Setup, Label Content Routing Config, Label Printing Routing Setup, and Label Printing Routing Config	page 33 page 35 page 37 page 39
	Added text for a new copy function in Label Format Maintenance	page 24
April 2019/version 3.1.2	For revision 2 of this user guide, updated the Version and Pause Between Label Generation fields in Label Format Maintenance.	page 24
April 2019/version 3.1.2	For revision 1 of this user guide, corrected paths used to import labels and related data.	page 69
March 2019/version 3.1.2	Rebranded for version 3.1.2.	--
	Added new chapter to import and use LPS out-of-the-box labels as templates.	page 67
	Added text for new Value frame that displays to hold up to 500 characters in Ad Hoc Label Generation.	page 63
	Added text for new Value frame that displays to hold up to 500 characters in Reprint Labels/Ad Hoc Generation.	page 55
March 2019/version 3.1.1	Rebranded for December 2018 version 3.1.1.	--
	Added information on deleting label data without orphan records remaining in Label Format Maintenance.	page 22
	Added new functionality to copy label data in Ad Hoc Label Generation.	page 63
	Removed appendix with entity diagram.	--
September 2018 /version 3.1	Rebranded for version 3.1	--
	Added stipulation for deleting a queue in Print Queue Maintenance and for updating history records in Label Request History.	page 15 page 56
	Added additional field information for following fields: <ul style="list-style-type: none"> Label Type field in Label Maintenance Printer field in Label Generation 	page 40 page 63
March 2018/version 3.0	Rebranded for version 3.0	--
	Specified illegal characters for Label Format Maintenance	page 21
	Made changes to Label Generation program.	page 63
	Added new Delete Label Format History (36.13.16.4.13) program.	page 22
September 2016/version 2.0.1	Rebranded for version 2.0.1.	page 53
	Added new feature to delete setups from the header frame in Label Content Routing Setup.	page 31
	Added information on entering blanks and quotation marks in fields.	page 43
March 2016/version 2.0	Added new functionality to reprint labels from the Label Print Status Collection.	page 53

Date/Version	Description	Reference
	Added new functionality to automatically delete old labels through the Delete Labels Before Month field in Label Format Maintenance.	page 22
	Added missing information for Manual Label Reprint (36.13.16.2.14)	page 62
	Added missing additional frame information to Auto Release Startup (36.13.16.2.16).	page 64
	Updated screenshots for several setup and processing programs, corrected outdated descriptions for programs, program processing, and fields throughout chapter 2 and chapter 3.	page 11 and page 51
	Reorganized the flow of Chapter 3 for improved information flow.	page 51
April 2015/version 1.2	Added new program for making calls to the label print services.	page 45
	Added information for making calls to Label Printing Services without QAD ICT.	page 45
	Added new Label Print Status Collection.	page 60
	Added new Delete Labels before Month field to Label Printing Services program to auto-clean Label Printing Services data.	page 22
	Added new stock fields and new copy function to Label Template Maintenance.	page 19
	Added new readability to Label Request History.	page 56
	Added new Label Printing Utility menu (36.13.16.4) and two new programs.	page 46
December 2014/version 1.1.30	Rebranded for version 1.1.30.0.	--
April 2014/version 1.1.20	Added new screenshots and supporting text for new fields in Label Printer Maintenance.	page 16
	Added information on a new Criteria Type Criteria Browse to peruse data based on the same-named maintenance program	page 43
	Added new browse collection information.	page 60
March 2014/version 1.1.14	Added information to use wildcards when setting up.	page 31
	Added new section for setup and configuration programs that let you specify label formats, datasets, and printers for advanced contextual routing setup to print for a specific purpose.	page 35
	Added information on printing through ICT middleware.	page 45
November 2013	Changed product name to Label Printing Services	--
October 2013/version 1.0	This is the first release of this user guide.	--



Overview

This chapter discusses the following topics:

Overview 2

Introduces label printing.

Features 5

Describes features and benefits when using label printing.

System Tokens 8

Describes software tokens that the system uses as a mechanism to define values in the data sent to the printers.

Label Printing Programs 9

Lists QAD EE label printing programs.

Limitations 10

Describes limitations for label printing.

Overview

The QAD Label Printing Services capability is integrated with the license plate inventory management and item serialization functions and lets you print labels directly from the various business process transactions to barcode printers.

Labels are a critical part of the process of tracking items and packages through production, through a warehouse, or from any receipt point to any delivery point. There are three main types of labels:

- Production labels are associated with items, and you typically create them when you create work orders.
- Package labels identify packages, and you generate them in the packing process.
- Shipping labels are typically customer-specific and printed during the shipping process.

Business Requirements/Needs

Previously, various modules within QAD EE provided the ability to print labels to Windows-based printers and through various menu functions. QAD EE modules that provide serialization capabilities, for example, had no separate QAD EE label print module to call when printing serial ID labels from the PC or from RF devices. There was also no means to set up the labels and printers so that specific label formats, printers, and so on could be configured.

QAD modules that provide serial ID functions, for example, had no separate QAD EE label print module to call when printing serial ID labels from the PC or from RF devices. Further, there was no means to set up the labels and printers so that specific label formats, printers, and so on could be configured.

Other QAD solutions also have label printing requirements. Solutions such as TMS, EAM, JIT/S, and QAD Warehousing require that users print barcode labels when processing orders with these modules. There were issues with the .NET UI, and polling programs frequently had to pause or have screens refreshed. Further, AppServers were not in place for a program's print request startup, connecting, processing, or polling. There was no database repository for print transactions. In short, there was no built-in label or barcode printing process and management system in place within QAD EE.

Finally, there was no means within QAD EE to interact with third-party label-printing or label-design software. Typical third-party label-design software generates the actual barcode representation for the printer. The label-design software usually offers many different syntax types such as ZPL, XML, CSS, IPL, and flat text files as well as control codes for specific barcode printers. There was no way for QAD programs to make use of the syntax templates provided by the third-party label-design software packages. Further, there was no record of the actual barcode label except in the barcode label printer system; however, this did not cover situations where a third-party barcode label software was in use.

Solution

The solution provides support for QAD EE users who use third-party barcode label printing software. A new Label Printing menu (36.13.16) now resolves label-printing requirements. Two main menus—Label Printing Configuration (36.13.16.1) and Label Printing Operations (36.13.16.2)—have several programs and functions that provide:



- Barcode label printing from QAD modules
- Configuration of label printing
- Support for both barcode printers and PC printers
- Label format design and modification capabilities
- Label inquiries and label reprint capabilities
- label printing
- Integration with third-party label-design software

Programs within the configuration menu let you set up the following to support the seamless printing of barcode labels:

- Printers
- Label types
- Label format definition
- Label template definitions
- Variable data field definitions
- Print queues and destinations
- Label restrictions

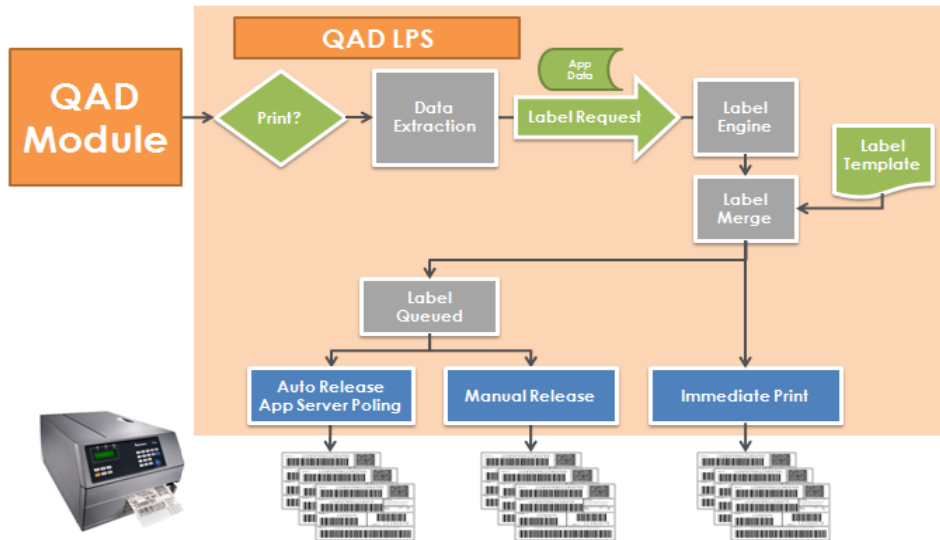
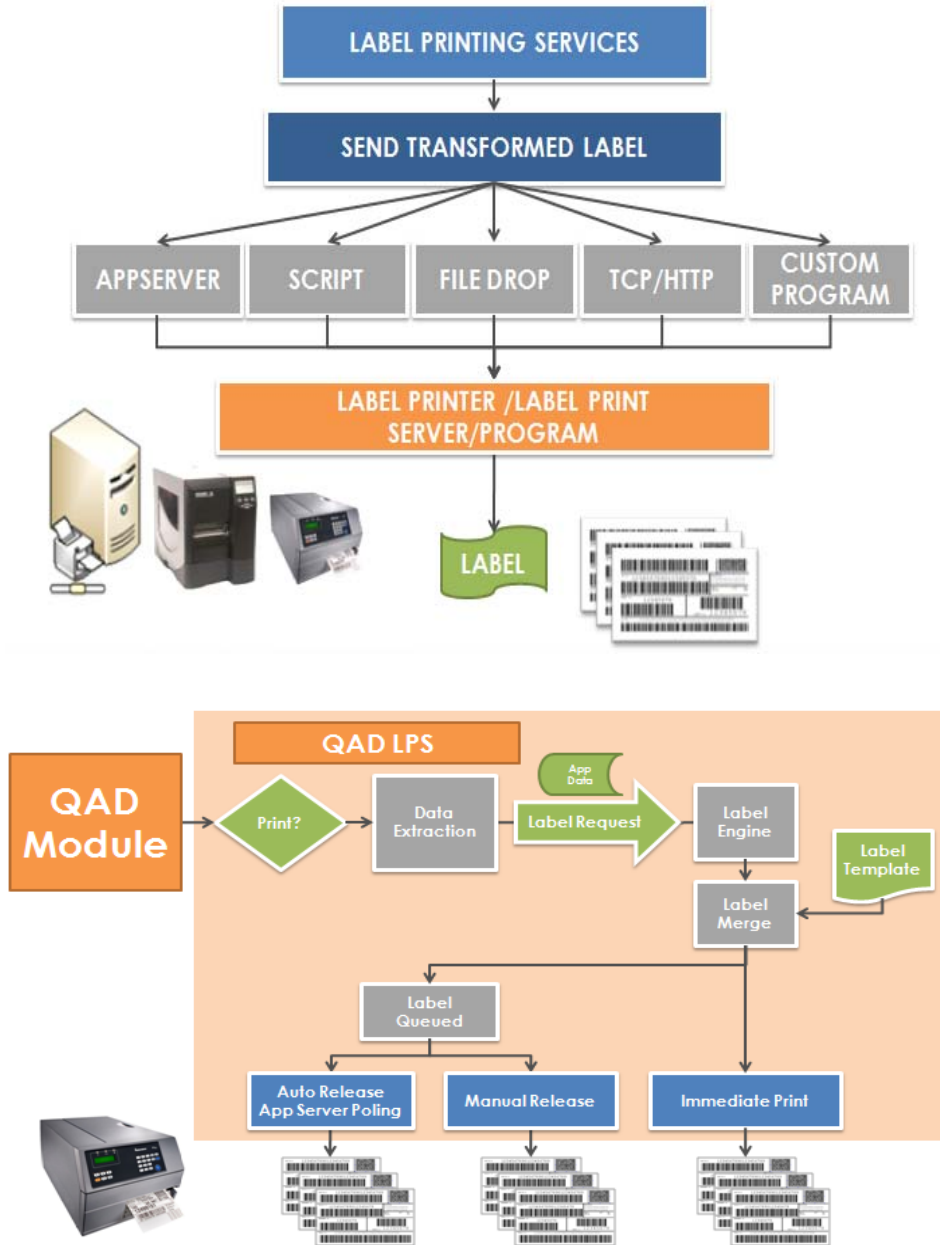
The label-printing module can interact with third-party label-design and printing software or directly with printers. It lets you print on PC (Windows) printers, too.

QAD EE applications can use Label Printing Services to generate and print barcode labels. Generating and printing barcode labels is the combined responsibility of the QAD Label Printing Services and external printers and printer software. You use the added label programs for configuring, maintaining, or reporting label printing functions. Providing a common label printing service extends consistent barcode label functionality across QAD applications.

Figure 1.1 shows the process flow within the Label Printing Services. It shows the Label Printing Services communication with the printer first, then the process within Label Printing Services.

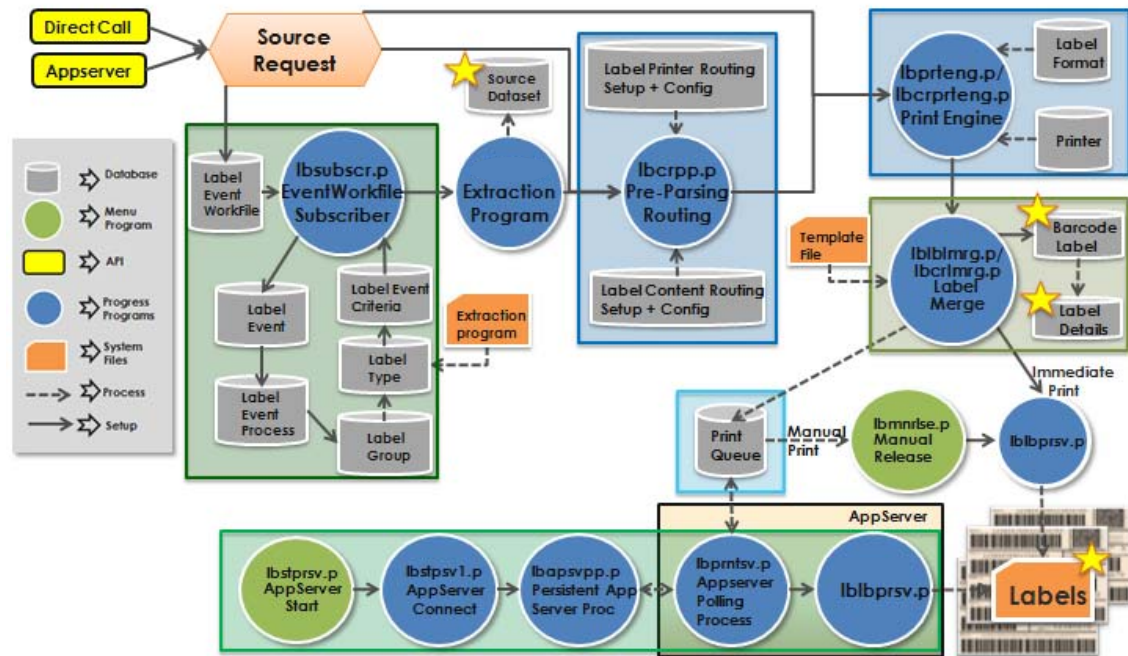
4 QAD Label Printing Services User Guide

Fig. 1.1
Label Printing Services Process Flow



The following graphic depicts the Label Printing Services detailed architecture involved with processing print requests.

Fig. 1.2
Label Printing Services Detailed Architecture



Features

Using QAD Label Printing Services, you can:

- Use and edit third-party label templates that are merged into QAD Label Printing Services.
- Receive and process print requests from various QAD EE programs.
- Print labels immediately, queue labels to be released manually, or release by a polling process using an AppServer.
- Monitor and track label printing data.
- Configure:
 - Connections and syntax types
 - Templates
 - Label formats and label mapping
 - Imported datasets for mapping and routing
 - Basic contextual routing or advanced contextual routing to get the printer and label format
 - Which applications are allowed to send requests to print labels
 - AppServer connections for queued labels
 - Printers
- Export label, printer, routing setup data, point-of-print, and extraction setup from Label Printing Services to the environment of your choice.

Barcode Labeling Service

QAD Label Printing Services provides full-featured barcode label functionality without burdening QAD applications. The barcode architecture consists of two parts:

- UI programs that let you create and maintain the tables required to configure the Label Printing Services
- Service-based applications that generate, track, and print barcode labels

You can create a specific printer for label printing, You can use new programs in the Label Printing menu to create groups of labels to be processed together or to separate labels into groups to be queued and printed at a later time.

Out-of-the-Box Labels as Templates

QAD provides a set of pre-designed labels that you can use as templates for your own labels. You first import the labels into your system. Once imported, you use the labels as templates.

Information in Chapter 4, “Using Labels as Templates,” on page 67 includes graphics of each label and a breakdown of the data on the label that corresponds to the Label Printing Service program, event, field, and so on that configures the label data.

Third-Party Templates

You can use templates from outside label design software within QAD Label Printing Services functions. The system generates the actual barcode representation for the printer. There are many different syntax types that you can use to design label templates, including:

- XML
- CSS
- PAS
- ZPL
- IPL
- Flat file
- Others

Fig. 1.3
Label Template Syntax Examples

<p>ZPL</p> <pre> ^XA ^FT50,100^A0N,75,75^FD tt-Data.company ^FS ^FT50,250^GB700,1,3^FS ^FT150,300^A0N,40,40^FD tt-Data.addr1 ^FS ^FT50,500^GB700,1,3^FS ^BY7,10,200^FT75,750^BC,,N,N^FD tt-Data.nbr ^FS ^FT300,800^A0N,40,40^FD tt-Data.nbr ^FS ^XZ </pre>	<p>IPL</p> <pre> ?H051;o152.074;f0;c7:w1:h1:d3,STR:??H052;o152.085;f0;c7:w1:h1:d3,LOC1:?? H053;o000.103;f0;c0:w1:h1:d3,QTY:??H054;o000.113;f0;c0:w1:h1:d3,(Q):?? H055;o000.155;f0;c0:w1:h1:d3,PART NO.(P):??H056;o395.242;f0;c0:w1:h1:d3,ENG ALERT:??H057;o182.103;f0;c0:w1:h1:d3,LINE FEED LOC 2:?? H058;o000.000;f0;c0:w1:h1:d3:??H059;o360.274;f0;c7:w1:h1:d3,CONTAINER:?? H060;o507.256;f0;c7:w1:h1:d3,GROSS WGT:??H061;o000.083;f0;c0:w1:h1:d3,MADE IN:??H062;o000.256;f0;c0:w1:h1:d3,SUPP (V):??H063;o000.355;f0;c0:w1:h1:d3, SERIAL NO (M):?? </pre>
<p>Software PASS file</p> <pre> *FORMAT,DHKTest.lwl *JOBNAME,SampleJob002 *QUANTITY,1 *PRINTNUMBER,1 so_mstr.so_cust, ft_so_mstr.so_cust sod_det.sod_part, ft_sod_det.sod_part sod_part, ft_sod_det.sod_part *PRINTLABEL </pre>	<p>XML Software</p> <pre> <?xml version="1.0" standalone="no"?> <!DOCTYPE labels SYSTEM "label.dtd"> <labels_FORMAT=SAMPLE.ZPL_QUANTITY="1"_PRINTERNAME="Printer1" _JOBNAME="LBL101"> <label> <variable name="SHIP_FROM"> ship.shipfrom </variable> <variable name="SHIP_TO"> ship.shipto </variable> <variable name="SHIPPER"> ship.shipid </variable> <variable name="SHIP_DATE"> ship.shipdate </variable> <variable name="CONTAINER"> ship.container.containerid </variable> <variable name="ITEM"> shipitem.itemnumber </variable> <variable name="ITEM_QTY"> shipitem.itemqty </variable> <variable name="BARCODE_ID"> serials.labelserialid </variable> </label> </labels> </pre>
<p>Software CSS file</p> <pre> *PRINTNUMBER;,*QUANTITY;,*FORMAT;,*DUPLICATES;,"so_mstr.so_cust","sod_det.sod_part","sod_part" 1,1,DHKTest.lwl, ft_so_mstr.so_cust , ft_sod_det.sod_part ,ft_sod_det.sod_part </pre>	

Configure Label Printing Services

You can use the new Label Printer Connection Maint (36.13.16.1.1) to set up connection details for label printing. When you set up details, the setup record determines which fields the system uses in a printer detail table for connection to the actual printers or printer servers. Available connections that you specify include:

- File drop
- TCP direct connection
- Scripts
- Program pass-through (passing context to another program)
- Program on AppServer (passing context to another program on the AppServer)
- TCP HTTP headers (uses HTTP headers to send the actual label to the printer)

Configure Syntax Types and Format

Syntax types group the *syntax*, which is the segmentation of relationships between structural elements used by the printer to interpret what is to be printed. The different syntaxes created are based upon third-party software, printers, and segments in use by various syntax types that support different label printing software. You can use Label Printer Syntax Type (36.13.16.1.2) to configure the various types.

The system combines the syntax and label format to make a specific combination to find the specific label detail to use when generating the contents of the labels. You use Label Format Maintenance (36.13.16.1.9) to set up a generic name for the label and specify label attributes. The combination of the label format and the syntax is a unique combination when determining the label printing capabilities. The same label that the system prints on two different printers can have different syntax for each (for example, XML and native barcode values); however, you define the label format only once.

Monitor and Track Label Printing Data

You can use any of the following to monitor and track label printing status:

- Label Request History (36.13.16.2.1)
- Label Queue Monitor (36.13.16.2.3)
- Label Maintenance (36.13.16.2.5)
- Label Print Status Collection (see “View Label Print Status and Reprint Labels” on page 53.)

System Tokens

The system uses tokens as a mechanism to define values in the data sent to the printers. When you create templates from third-party barcode design software, you can use Label Printing Services functions to replace variable that you want populated with data. These are variables such as field names or values that you recognize. The variables are surrounded by a variable separator configured in Label Template Maintenance.

Tokens start and end with an @ symbol. Tokens are used to get information from the system that may not be available to the printing engine as a default. There are two types of tokens:

- System tokens

The label engine assigns values for these tokens. The system tokens are specific values that the system uses for processing. System tokens include the following:

- @Copies@

This is the number of copies to print.

- @ExternalPrinterID@

This is a value defined in the printer maintenance screen.

- @JobName@

This is a request ID.

- @SystemDate@

- @SystemTime@

- @LabelFormatID@

This is a value defined in Label Format Maintenance.

- Data tokens

These tokens are available for users and can be associated with the `key` values within a dataset. Data tokens include the following:

- @Domain@

This is the Domain field, free-form defined by the user and stored in the label domain field.

- @Purpose@

This is the Purpose field, free-form defined by the user and stored in the label purpose field.

@labelKey1@ through @labelKey3@

These are label Key1 through label Key 3 fields, free-form defined by the user and stored in label key1 through label key 3 fields.

@srcApplication@

This is the source application, free-form defined by the user and stored in the label key field.

Label Printing Programs

The following table lists QAD EE label-printing programs:

Table 1.1
QAD EE Label Printing Programs

Menu	Program Label	Program Name
36.13.16	Label Printing Menu.....	
36.13.16.1	Label Printing Configuration....	
36.13.16.1.1	Label Printing Connection Maint	lbcontmt.p
36.13.16.1.2	Label Printer Syntax Type Maint	lbsntyymt.p
36.13.16.1.3	Printer Model Maintenance	lbprtmmt.p
36.13.16.1.4	Print Queue Maintenance	lbprqumt.p
36.13.16.1.5	Label Printer Maintenance	lbprntmt.p
36.13.16.1.7	Label Template Maintenance	lblbtpmt.p
36.13.16.1.9	Label Format Maintenance	lblbfmmt.p
36.13.16.1.11	Label Dataset Import	lbdsimp.p
36.13.16.1.12	Label Dataset Maintenance	lbdsrmt.p
36.13.16.1.13	Label Application Configuration	lblbapmt.p
36.13.16.1.14	Label Printer Router Browse	lbb018.0
36.13.16.1.15	Label Content Routing Setup	lbrstupmt.p
36.13.16.1.16	Label Content Routing Configuration	lbrcfgmt.p
36.13.16.1.17	Label Printer Routing Setup	lbprstmt.p
36.13.18.1.18	Label Printer Routing Config	lbprcfmt.p
36.13.16.1.19	Label Mapping Functions	lbfuncmt.p
36.13.16.1.20	Label Mapping	lbmapmt.p
36.13.16.1.21	Label Format Router Browse	lbb021.p
36.13.16.2	Label Printing Operations...	
36.13.16.2.1	Label Request History	lbrqshmt.p
36.13.16.2.3	Label Queue Monitor	lbpbqmt.p
36.13.16.2.5	Label Maintenance	lblblbmt.p
36.13.16.2.6	Label Print Status Browse	lbb014.p

Menu	Program Label	Program Name
36.13.16.2.7	Label Reprint Maintenance	lblbrpmt.p
36.13.16.2.13	Label Manual Release	lbnmrlse.p
36.13.16.2.14	Manual Label Reprint	breprint.p
36.13.16.2.16	Label Auto Release Startup	lbtprsv.p
36.13.16.2.19	Label Generation	lbadhoc.p
36.13.16.2.29	Label Reprint/ Generation	lbrxrepad.p
36.13.16.3	Label Extraction Configuration Menu	
36.13.16.3.1	Label Type Maintenance	lbltypmt.p
36.13.16.3.4	Label Group Maintenance	lblgrpmt.p
36.13.16.3.13	Label Event Maintenance	lbevtmt.p
36.13.16.3.14	Label Event Processing	lbpgrpmt.p
36.13.16.3.15	Label Event Type Criteria Maintenance	lbevtymt.p
36.13.16.3.16	Label Event Type Criteria Browse	lbb029.p
36.13.16.4	Label Extraction Configuration Menu...	
36.13.16.4.1	Label Config Import/Export	lbcrgie.p
36.13.16.4.2	Label Extraction Import/Export	lbexcfie.p
36.13.16.4.3	Delete Label Format History	lbdellabelhist.p
Browse Collections		
.NET UI only	Label Print Status Collection	
.NET UI only	Label Event Type Criteria Browse	
.NET UI only	Label Printer Router Browse (view only)	
.NET UI only	Label Format Router Browse (view only)	

Limitations

You must configure the label formats in the setup screen. The label formats you define determine what the system searches on in the searchable fields in the labels table.

When you define the label template in the label engine, you also define which fields from the label data the system uses as keys for looking up the label after the system prints the label. If you do not define fields, the labels do not use referenceable data to find them again.

Having the system pass information to the label applications (default configuration) does not require you to define traceable keys. You only set this up when you want to change the system default. This makes the label definition less complex.

Label Print Setup

This chapter discusses the following topics:

Setup Overview 12

Introduces the setup process for label printing.

Set Up Label Printing Master Data 13

Tells you how to create a connection to a valid AppServer when you need labels released by the automated polling-release functions.

Set Up Printers 14

Describes various EE programs that you use to set up label printing.

Define Label Formats 18

Describes several programs to use to set up label formats.

Set Up Contextual Routing 30

Describes setup and configuration programs that let you specify datasets, label formats, and printers for advanced contextual routing setup.

Set Up Contextual Printer Routing 35

Describes setup and configuration programs that let you specify label formats, datasets, and printers for advanced contextual routing setup.

Set Up Label Extraction 40

Provides information to use the programs in the Label Extraction Configuration menu (36.13.16.3).

Import/Export Label Setup Data 46

Tells you how to use the programs in the Label Printing Utility Menu (36.13.16.4) to extract printer, label, and other setup data to another environment.

Delete Stranded Work Flows 48

Tells you how to delete stranded or unwanted label event work flow records.

Set Up Labels for QAD Serialization 48

Describes the setup and process to print QAD Serialization labels.

Data Collection Transactions 49

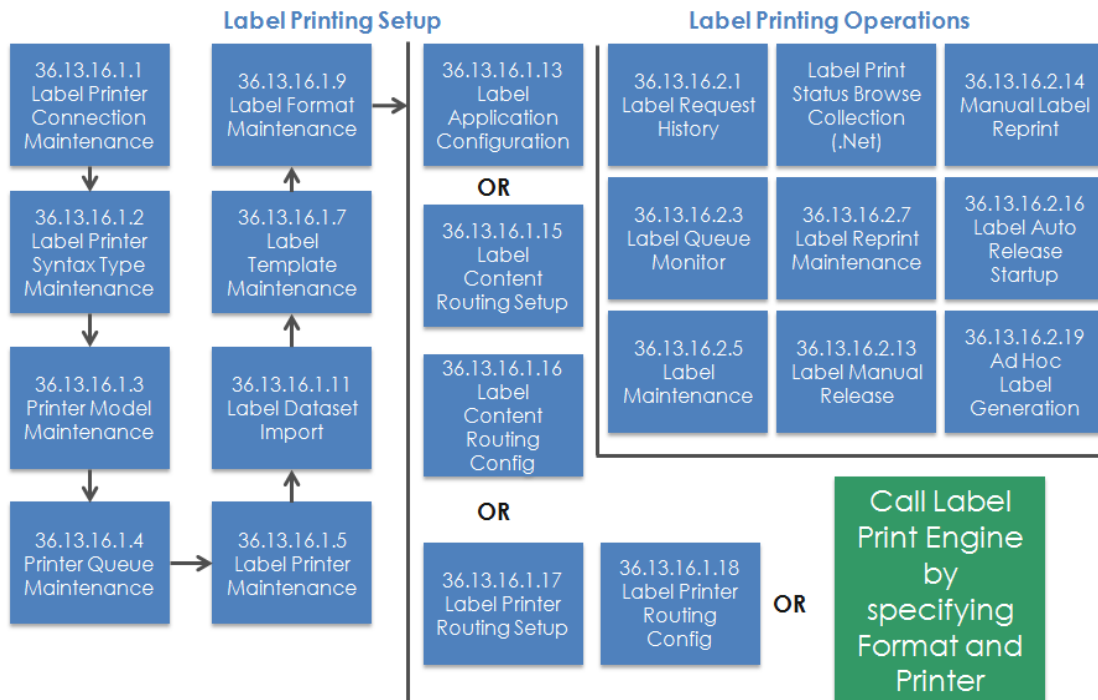
Provides a linked reference to the latest Automation Solutions: Data Collection transactions.

Setup Overview

The following topics describe the setup for label printing functions within QAD EE. Setup of standard QAD EE components, such as sites, items, orders, entities, and so on, is not covered in the following topics, unless specific label printing fields exist within the QAD EE maintenance programs that let you set up these basic QAD EE components. See the QAD EE *Master Data* and *Manufacturing* user guides for setup of QAD EE components.

The following setup flow provides a general approach to setting up label printing.

Fig. 2.1
Setup Flow



A minimal configuration of Label Printing Services requires that you specify the template from third-party label design software or use your own template, and set up connection types, syntax types, printer models, template maintenance, format maintenance, and printer maintenance.

More advanced setup includes the dataset import, application maintenance, and mapping functions.

The following topics present label printing setup information:

- “Set Up Label Printing Master Data” on page 13
- “Set Up Printers” on page 14
- “Define Label Formats” on page 18
- “Configure Applications” on page 29

Set Up Label Printing Master Data

When you need labels released by the automated polling-release functions, you must create a connection to a valid AppServer.

Note Every AppServer with an initiated polling process uses an agent; you should be aware of the number of available agents on the AppServer.

The following topics discuss programs that you can set up for label printing by site in QAD EE.

Connect to an AppServer

Your system administrator can access AppServer Maintenance (36.19.1) to define the information needed to connect to a Progress AppServer.

You can specify a set of standard connection parameters used to connect to this server. Optionally, you can also define server-specific parameters required by the AppServer.

Important You must enter `lbPrintServer` as the Application Service; otherwise, the connection does not work.

Fig. 2.2
AppServer Service Maintenance (36.19.1)

The screenshot shows a window titled "AppServer Service Maintenance" with a menu bar containing "Go To", "Actions", "Copy", "Print", "Preview", and "Attach". The main area contains the following fields:

- Service Name: lbPrintServer
- Description: Print server for Label Printing Services
- Application Service: (empty text box)
- IP Address or Host Name: (empty text box)
- Port Number: (empty text box)
- Parameters: (empty text box)

Service Name. Enter a name to identify this application server.

Description. Optionally enter a description of the application server.

Application Service. Enter the name of the Application Server defined in the `ubroker.properties` file during configuration of the AppServer.

IP Address or Host Name. Enter the IP address or host name used as the `-H` parameter when connecting to this application server. This is the IP address or host name of the machine on which the application server is running. If the AppServer is running on the same machine as your QAD database, enter the local host.

Port Number. Enter the port number used when connecting to this application server.

- If you are running a Progress name server, enter the name server port number. The default value is 5162.
- Otherwise, enter the port number on which the AppServer is running.

Parameters. Optionally enter any other parameters required when connecting to this application server.

Set Up Printers

The following topics describe programs you use in the printer setup.

Label Printer Connection Maintenance

Use Label Printer Connection Maintenance (36.13.16.1.1) to set up connection details. The data in this program serves as a template for printer detail setup. Setting fields in this program determines which fields the system uses in the printer detail table.

Fig. 2.3

Label Printer Connection Maintenance (36.13.16.1.1)

Connection Type. Indicate a unique ID that identifies connections in other tables.

Connection Type Desc. Optionally, enter a description of the connection type.

Use Default Destination. Indicate Yes when the destination directory can be updated in the printer detail connection setup screen.

Use Script. Indicate Yes when script field can be opened in the printer detail screen.

Use TCP protocol. Indicate Yes when the TCP settings can be updated in the printer detail setup screen.

Use HTTP Headers. Indicate Yes when the TCP settings incorporate an HTTP header before sending the content of the label.

Use App Server. Indicate Yes for the AppServer to send the content to an OpenEdge program on the AppServer.

Label Printer Syntax Type Maintenance

Use Label Printer Syntax Maintenance (36.13.16.1.2) to group the syntax that is used to send to printers. The system uses label-printing syntax—such as XML, CSV, PAS, ZPL, IPL—in printer models to determine which printers use which syntax type and eventually when the label prints on that printer.

Several different syntaxes are available based upon third-party software, printers, and whether a third party is parsing the file and sending to the printer or if the printer is using a native sending of the data.

Fig. 2.4
Label Printer Syntax Maintenance (36.13.16.1.2)

Syntax Type. Indicate a unique grouping of syntax used to generate the label.

Description. Optionally, enter a description of the syntax.

Printer Model Maintenance

Use Printer Model Maintenance (36.13.16.1.3) to set up defaults for printer types. The system uses printer models for default values for the printer detail connection screen. The models determine the default value of connections that the system uses to communicate with the printer. This also allows for configuration of similar printers to only be set up once.

Fig. 2.5
Printer Model Maintenance (36.13.16.1.3)

Model ID. Indicate a unique ID to describe the printer model.

Description. Optionally, enter a printer model description.

Stock Height. Enter the size of the label height.

Note Currently not implemented other than as a reference.

Stock Width. Enter the size of the label width.

Note Currently not implemented other than as a reference.

Connection Type. Enter a valid connection type defined in Connection Type Maintenance. This is the default value for the printer detail connection screen.

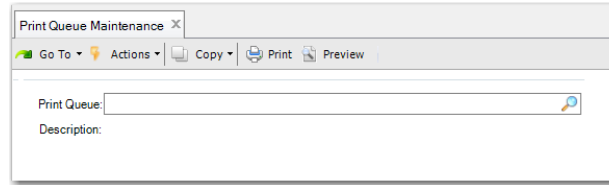
Print Queue Maintenance

Use Print Queue Maintenance (36.13.16.1.4) to create groups of labels to be processed together or to separate labels into groups to print at a later time. The printer queue acts as an identifier that allows the printing engine to determine if data prints immediately or later by the manual label release or the automated label release.

Note You cannot delete a queue in this program when the queue is in a generated state in Label Maintenance (36.13.16.2.5). When you try, the system displays an error, informing you that a label queue record exists and the deletion is not possible.

Fig. 2.6

Print Queue Maintenance (36.13.16.1.4)



Print Queue. Indicate a unique ID that represents a holding location for labels to be held in sequence until a process releases the labels to the printer.

Description. Optionally, enter a description of the print queue.

Label Printer Maintenance

Use Label Printer Maintenance (36.13.16.1.5) to:

- Create a specific printer.
- Set up connections for the printer.

In the first frame, you set up the printers and their connection definitions. Printers can be virtual printers (middleware print servers) or actual physical printers. Use the first screen to set up the combination of the printer model and syntax the system uses to generate labels on that printer.

In the second frame, you specify printer connection details that determine how the system communicates with the printer. The values that you can update are based on the connection type defined in the first frame. The system displays the default connection type, which you can edit.

Fig. 2.7
Label Printer Maintenance (36.13.16.1.5)

The screenshot shows two overlapping windows from the 'Label Printer Maintenance' application. The top window, titled 'Label Printer Maintenance', shows the configuration for 'Printer: testprinter'. It includes fields for Description (testprinter), External Device Printer (test), Format Prefix (ZP), Format Suffix (ZS), Printer Model (FileDrop), Syntax Type (Test), and Print Queue. The bottom window, titled 'Printer: lpr01', shows configuration for a different printer. It includes fields for Connection Type (contp01), Send Program, AppServer Name, Dest Dir (/tmp), Script, Server (167.3.9.62), URL, Port (2812), and a checked 'Use HTTP Headers' checkbox. An arrow points from the top window to the bottom window.

Printer. Indicate a unique ID that identifies the printer.

Description. Optionally, enter a description of the printer.

External Device Printer. Enter the external third-party application printer. This is tied to the system token `@externalPrinterID@`, which is replaced in the template when the label is generated. For more information, see “System Tokens” on page 8.

Format Prefix. Enter characters to attach to the label as a prefix. For example, if you enter ZP, and the labels are set up as numbers starting from 00001, the system reports and prints ZP-00001, and so on.

Format Suffix. Enter characters to attach to the label as a suffix. For example, if you enter ZS, and the labels are set up as numbers starting from 00001, the system reports and prints 00001-ZS, and so on.

Printer Model. Enter a valid printer model defined in Printer Model Maintenance (36.13.16.1.3).

Syntax Type. Enter the valid syntax type defined in Label Printer Syntax Type Maintenance (36.13.16.1.2).

Printer Queue. Enter a valid print queue defined in Print Queue Maintenance (36.13.16.1.4) or leave blank for immediate printing.

Connection Type. This field defaults from the printer model defined in the first frame.

Send Program. Enter the program to which the system delivers content. If the AppServer name is configured and is valid, the system sends content to this program for the AppServer to process. The signature of the program that receives the generated label is as follows:

- Input labelDetData as character (the entire labels that generate)
- Input Buffer labelConnDet as buffer

- Output `op_wasSuccess` as logical

AppServer Name. Enter a valid AppServer name defined in AppServer Server Maintenance (36.19.1). This runs the program defined in the send program field on the AppServer.

Dest Dir. Specify a directory in which the system creates files.

Script. Specify one of the following scripts for the system to use. This field cannot be blank when Use Script in Label Printing Connection Maintenance (36.13.16.1.1) is set to Yes.

%F - File name

%D - Date and Time

%R - Request ID

%L - Label ID

%M- Label Format ID

%C - Copies

%N - Domain

%S - Source Application

%P - Purpose

%1 - Label Key 1

%2 - Label Key 2

%3 - Label Key 3

The scripts let the system use command line values while calling the script. You must have a file created to use the scripts.

Server. Specify either the IP address or the name of the host server to use for TCP connections.

URL. The URL used to send information to a web server, when used. This field cannot be blank and is only editable when Use HTTP Headers is set to Yes. When you select a different connection type from the drop-down, the system clears fields that are not applicable to the new connection type.

Port. Specify the port on the server for TCP communications. Most web servers are 80 or 8080; however, this lets the system print directly on printers using this port.

Use HTTP Headers. Indicate Yes to send information over HTTP requests when using TCP as a medium for connection.

Define Label Formats

You use several programs to set up label formats. The following topics describe the programs you use:

- “Label Template Maintenance” on page 19
- “Label Format Maintenance” on page 21
- “Label Dataset Maintenance” on page 26
- “Label Mapping Functions” on page 27



- “Label Mapping” on page 28

Label Template Maintenance

Use Label Template Maintenance (36.13.16.1.7) to

- Maintain the label template.
- Import the template during the maintenance of the records.

The templates are based on a specific syntax and need to be set up as such.

There are several frames available in this program. In the first frame, enter the printer syntax type and template name, then press Enter.

Important The system displays the contents of the template; however, you cannot update the contents.

Press Enter again to display the variables that were found in the template file during the import. Scroll through the variables, updating the descriptions. You cannot change the variable name; however, you can update the variable description.

Important Each time you go through the frame, the system reimports the file.

To understand content, note the following:

Anything starting with an @ and ending with an @ is a system token. These are used to get information from the system that may not be available to the printing engine as a default. The available tokens in the file are as follows:

```
@SystemDate@ = System Date
@SystemTime@ = System Time
@ExternalPrinterID@ = value defined in the printer maintenance screen
@JobName@ = request ID
@srcApplication@ = Source Application - free-form defined by user -
stored in the label key field
@Domain@ - domain field - free-form defined by user - stored in the
label domain field
@LabelFormatID@ - value defined in the Label Format Maintenance
@Purpose@ - Purpose field - free-form defined by user - store in the
label purpose field
@labelKey1@ -label Key1 field - free-form defined by user - stored in
label key1 field
@labelKey2@ -label Key2 field - free-form defined by user - stored in
label key2 field
@labelKey3@ -label Key3 field - free-form defined by user - stored in
label key3 field
```

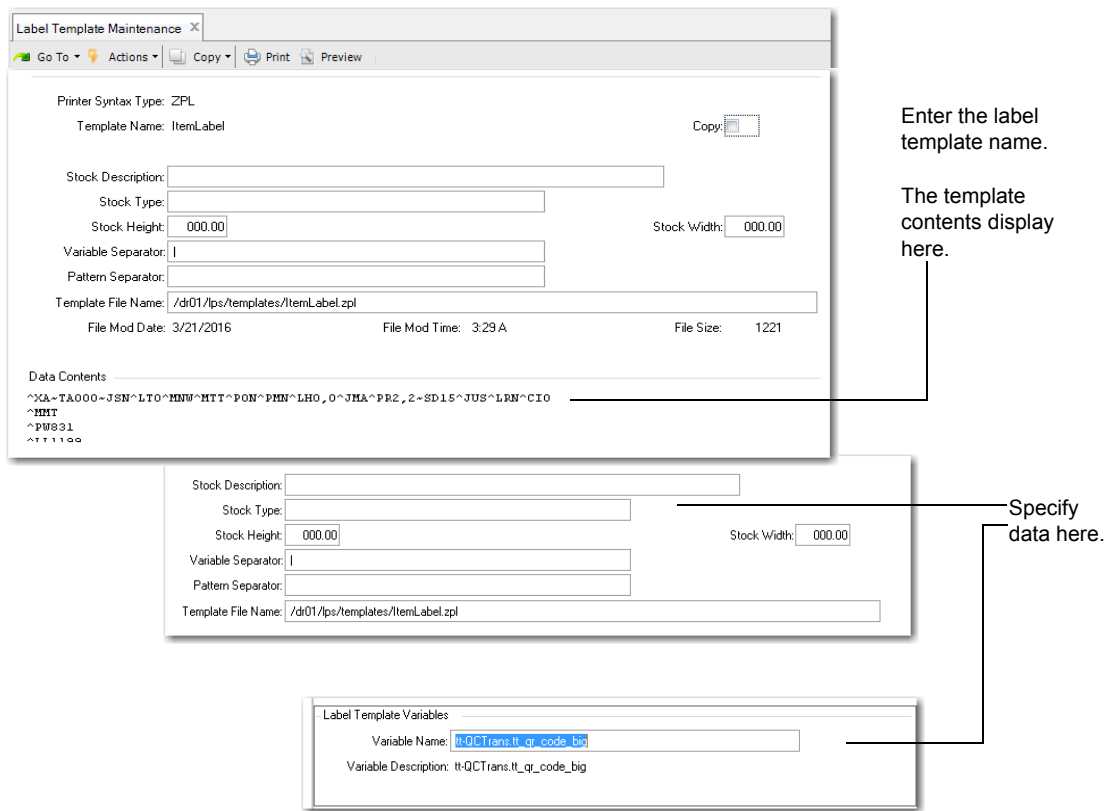
Copy Function

The program includes a Copy function. When you set the Copy field to copy a template, a floating frame lets you modify the following fields:

- Template Name
- Stock Description
- Stock Type
- Stock Height
- Stock Width

Once the system completes the copy, you return to the original screen in .NET UI or press F4 in character mode. You can continue to copy an existing setup and just change Stock Description, Type, Height, and Width to base the new template on a current template.

Fig. 2.8
Label Template Maintenance (36.13.16.1.7)



Printer Syntax Type. Indicate a valid syntax type defined in Label Printer Syntax Type Maintenance (36.13.16.1.2).

Template Name. Specify a user-defined, unique ID for the template.

Copy. Set to Yes to copy a template; a floating frame lets you modify fields. You can continue to copy an existing setup and just change Stock Description, Type, Height, and Width to base the new template on a current template.

Variable Separator. Enter a variable separator to identify the character that shows where the system places tokens. This should be a value in the template file that indicates the tables, fields, and system tokens.

Pattern Separator. Enter a separator that the system uses to update repeating patterns in single files. When there is a segment of data that is repeating and a single file houses several different labels, the system uses this separator to indicate the separation. If there are multiple labels generated, then the single output has all the labels created inside of it.

Stock Description. Enter a description of the stock.

Stock Type. Enter the stock type.

Stock Height. This field displays the height of the stock being printed.

Stock Width. This field displays the width of the stock being printed.

Template File Name. The system displays the template name.

File Mod Date. This field is display only and indicates the modification date of the file when the file was last imported.

File Mod Time. This field is display only and indicates the modification time of the file when last imported.

File Size. This field is display only and indicates the size of the file when it was last imported.

Template File Name/Description. Enter the name of the template file and, optionally, a description.

Variable Name. The variable name and description come from the import of the template defined in the template file name. The value between the | signs represents both the name and the description. To make this value a user-readable/understandable value, update the value in this field.

Label Format Maintenance

Use Label Format Maintenance (36.13.16.1.9) to create a generic label format name and specify label attributes.

Use one frame to specify the generic label format and a second to specify label format details that connect the template to the syntax of the printer. The syntax and label format combine to make a specific combination to find the specific label detail that the system uses to generate the label contents. When the system connects the syntax to the template, it uses either the merge program, the label map, or the default mapping to make the connection. When you set Tokens to Yes, a third frame displays.

As you enter data in frames, you can leave fields blank when they are not required. Also, the program validates illegal or invalid characters in file name prefixes and suffixes when the system generates file names and displays an error message when file name prefixes and suffixes contain a space or the following special characters:

,	:	#
<	"	!
>	[^
?]	*
/	{	(

,	:	#
;	})
'	`	+
=	\	

Delete Label Data

You can have the system delete label data by specifying the number of months of label data you want to preserve in Label Format Maintenance. The Delete Labels Before Month field lets you enter the number of months to preserve data. With this auto-cleaning function built into the Label Printing Services program, you do not need to run a separate delete/archive utility as the system deletes data belonging to the format before the date you indicate in the field.

So, when you specify the number of months of label data that you want to preserve, the system uses that number to calculate a date that serves as the beginning date for which it deletes label data. For example, today is 4/13/2019 and you set Delete Labels Before Month to 2. The system calculates a beginning date of 2/13/2019 and deletes all data prior to 2/13/2019.

The system also automatically deletes any old labels that fall before the date you enter as it generates new labels that are based on the format. For example, you enter a date that is one month from today in the Delete Labels Before Month field. Tomorrow you enter the system and you no longer have to delete old labels a month before the date. The system deletes the old labels as you create new labels based on the format. You can also delete a format when the format has either tokens or details.

You can run the feature when the system generates all labels and sends them for printing. Since the system stores the generated labels in Label Printing Services tables, performance can be affected over time with an accumulation of stored label data. This feature auto-cleans Label Printing Services-related transaction type tables, thereby improving Label Printing Services performance.

When you delete records from within this program, the system does not leave orphaned records in configuration tables that are related to label content routing or label printer routing. To copy records, see Copy Label Data.

Fig. 2.9
Label Format Maintenance (36.13.16.1 9)

The screenshot shows a web application window titled "Label Format Maintenance". At the top, there is a navigation bar with "Go To", "Actions", "Copy", "Print", and "Preview" options. Below this is a search bar for "Label Format" and an "Active" checkbox. The main area contains several configuration fields: "Label Desc", "Track" (checkbox), "Delete Labels before month", "Generate Files" (checkbox), "Tokens" (checkbox), "File Prefix", "File Suffix", "Copies", "Version", "Pause Between Label Generation", "Printer Syntax Type", "Merge Program", "Label Map", "Label Template", "Token Name", and "Dataset Field Name".

Label Format. Indicate a unique value that identifies the label format.

Label Description. Optionally, enter a label description.

Active. Indicate Yes to activate the label. When No, the label print engine does not print the label and indicates to the user that the label is inactive.

Track. Indicate Yes for the system to store labels in the label table.

When No, then label reprints are not applicable. The labels are temporarily created but are deleted after they are printed.

Delete Labels Before Month. Specify the number of months to preserve data or accept the default. The system uses the number to calculate a date that serves as the beginning date for which it deletes label data. The system validates this field for a numeric value.

When you enter a value, the system prompts you to confirm the data deletion. The confirmation includes the starting date of the deletion. When you respond with Yes, the system deletes label records and displays a message indicating that label records have been successfully deleted. When you respond with No, the system does not delete data and displays a message indicating that label records were not deleted. The deleted labels are not archived.

Generate Files. Indicate Yes to generate files. You must specify Yes when labels are using any connections that require a file to be generated.

File Prefix. Enter a value that the system adds to the front of the generated file name.

File Suffix. Enter a value the system appends to the file name; typically, this is the file extension.

Tokens. Set to Yes to enter tokens in a third frame.

Note Select Back to return to the second frame, then Next to display the Token frame.

In the Token frame (see Figure 2.9), you enter any programming tokens that describe functions (constants, identifiers, operators, reserved words, separators) that are in use in the label data syntax. When you enter tokens, a third frame displays data that comes from the class codes. Valid values are as follows:

- srcApplication
- Domain
- Purpose
- labelKey1
- labelKey2
- labelKey3

Copies. Enter the number of copies of the label to generate. You cannot enter a 0 (zero), and the default is 1.

Dataset. The dataset associated with the label format.

Version. This field shows the dataset version. You can edit this field for different values when there are different dataset versions available only when the Dataset field in this program is populated.

Pause Between Label Generation. Enter the second(s) to be used as the pause between the generation of each label. When you create a new record, the default value is zero (0) seconds.

Printer Syntax Type. Indicate a valid syntax type defined in Label Printer Syntax Type Maintenance (36.13.16.1.2).

Merge Program. Enter the name of the application that the system uses to merge the data from the incoming source to the label.

Label Map. Enter a valid map defined in Label Mapping (36.13.16.1.20). The merging occurs based on the setup of the map.

Label Template. Enter the name of the template to connect to the syntax when the system does not use either the merge program or the map. With just the syntax and the template defined, the generic label merging occurs.

Token Name. This display-only field shows the token name.

Dataset Field Name. Specify a dataset field to be linked with the indicated token.

Copy Label Data

You can copy an existing label format definition in Label Format Maintenance and name the copy to a new label format definition. You select Copy Label Format from the drop-down menu from the Action option to access the function.

The copy function is available when the following fields are enabled:

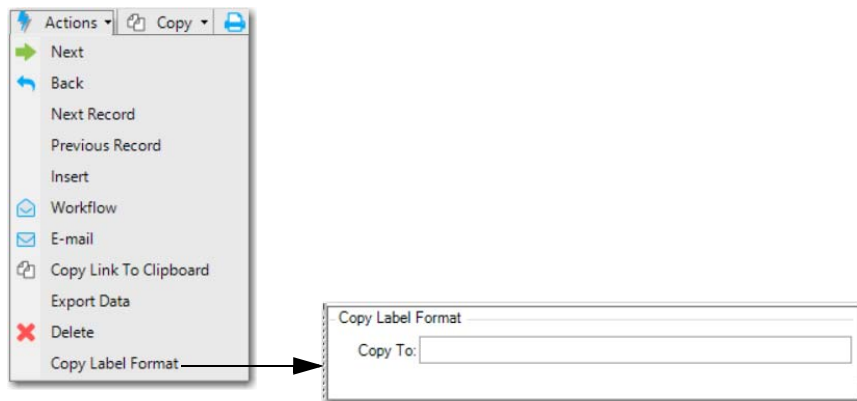
- Active
- Label Desc
- Track
- Delete Labels before month



- Generate Files
- Tokens

When the fields are enabled and you select the Copy Label Format action, the system prompts you to enter a new label format in the Copy To field. When the new label format already exists, the system displays a message. You must re-enter a new format label.

Fig. 2.10
Copy Label Format



You can return to the field that you were on when you invoked the Copy Label Format function by selecting to go back. The system informs you when the copy has completed and displays the new label format with the copied data.

Delete Label History Records

Use Delete Label Format History (36.13.16.4.13) to delete label history records for all label formats. When you use the utility, it prompts you to enter the number of days of history to preserve in the system.

Use the Days of History to Retain field to specify any number of days between 0 and 999; the default is 7. When you press Go, the utility deletes label history data that is not specified in the preservation days.

Use the Simulate field to simulate the deletion first; the system displays records it intends to delete. When you uncheck Simulate, then press Go, the system deletes the records.

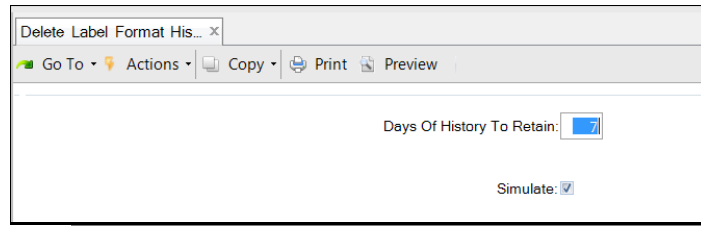
The utility uses the following formula to calculate the preservation date:

$$\text{Delete dates} = \text{add-interval}(\text{today}, (-1 * \text{<number of days to retain>}), \text{day})$$

Since the history dates are based on date and time, the system sets the delete dates time to 23:59:59:999 so that it can pick up all history records for that day. The system deletes the following tables:

labelMstr	labelDet
labelPrntQue	labelReprint
responseHstD	requestHist
requestHstRsp	sourceMsg

Fig. 2.11
Delete Label Format History (36.13.16.4.13)

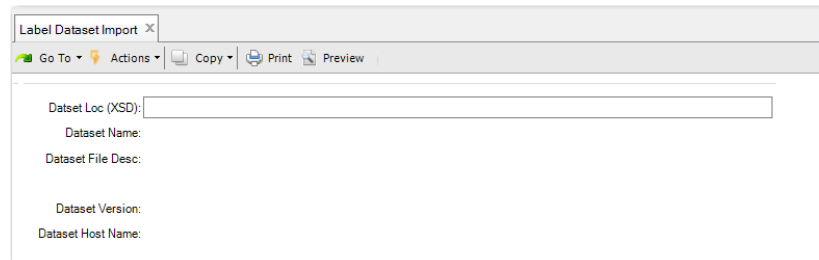


Label Dataset Import

Use Label Dataset Import (36.13.16.1.11) to define datasets that the system stores and uses to determine the incoming message structures. These can be generated from QXtend or other applications to let the label mapping engine determine the incoming data structure.

The dataset is an internal representation of the dataset that is coming from the calling application. This is usually stored in an XSD file for parsing.

Fig. 2.12
Label Dataset Import (36.13.16.1.11)



Dataset Loc (XSD). Indicate the file name as it exists in the operating system. As with all UNIX file names, uppercase and lowercase are distinct.

Dataset Name. Enter a user-defined name that identifies the dataset. The default is the name of the XSD file; however, you can edit it.

Dataset File Desc. Optionally, enter a description of the dataset.

Dataset Version. Indicate a dataset version as there can be multiple versions of the dataset.

Dataset Host Name. Indicate the machine in which the XSD is located.

Label Dataset Maintenance

Use Label Dataset Maintenance (36.13.16.1.12) to view and delete datasets.

Use the second frame to help maintain the datasets by viewing different versions of the dataset definition. You can see each version by using the up and down arrows on the version number.

Use the Display option to display the dataset on the screen in XSD format. To determine which tables and fields are available for use, enter Table in the display box to display the buffers and fields in the dataset.

Fig. 2.13
Label Dataset Maintenance (36.13.16.1.12)

Label Dataset Maintenance X

Go To Actions Copy Print Preview

Dataset Name: dsItemLabelType

Dataset File Desc:

Dataset Current Version: 1

Dataset Location (XSD): /home/demo-admin/dsItemLabelType.xsd

Version:

Display: XSD

```
<?xml version="1.0"?><xsd:schema
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns=""
xmlns:prodata="urn:schemas-progress-com:xml-prodata:0001"><xsd:element
name="dsItemLabelType"
prodata:proDataSet="true"><xsd:complexType><xsd:sequence><xsd:element
name="tt-ItemLabelType" minOccurs="0"
maxOccurs="unbounded"><xsd:complexType><xsd:sequence><xsd:element
name="pt_part" type="xsd:string" nillable="true" prodata:format="x(18)"
```

Version: 1

Display: Table

```
tt-ItemLabelType.pt_xsub_l1
tt-ItemLabelType.pt_xsub_std11
tt-ItemLabelType.pt_xsub_std1
tt-ItemLabelType.pt_xsub_t1
tt-ItemLabelType.pt_xtot_cur
tt-ItemLabelType.pt_xtot_std
tt-ItemLabelType.pt_yield_pt
tt-ItemLabelType.SourceAppRequestID
```

Dataset Current Version. Enter the name derived from the dataset contents. You cannot change the name.

Dataset File Desc. Optionally, enter a new description or edit the existing dataset description.

Curr Version. Enter the current version for the dataset. When the system receives a request to print a label, the system uses the current version to parse the dataset.

Dataset Loc (XSD). Specify the actual location of the file in the operating system.

Version. Select the dataset version to view.

Display. To view an XSD representation, specify XSD. To view a list of available dataset fields, specify Table.

Label Mapping Functions

Use Label Mapping Functions (36.13.16.1.19) to specify programs that access other data and perform additional tasks.

The system uses mapping functions during the mapping process to get data into the right format before merging it onto the label.

Use the first screen to identify the function and program. Use the second screen to define the parameters for the function.

Fig. 2.14
Label Mapping Functions (36.13.16.1.19)

Seq	Data	Parameter Name	Default Value
1	c	test	

Function Name. Indicate a unique name for the function.

Description. Optionally, enter a description of the function.

Program Name. Enter the program name as it exists on the disk. The system automatically generates a file when it does not exist. You can overwrite the file, so be careful when updating.

Seq. This is the numerical identifier that determines the sequence of the parameters that the system passes.

Data. This is the type of data that the function is passing. Examples are:

- C – Character
- R – Decimal (Real number)
- I – Integer
- L – Logical
- D – Date

Parameter Name. Enter a valid variable name for the OpenEdge program.

Default Value. Enter the default value you want the parameter to have.

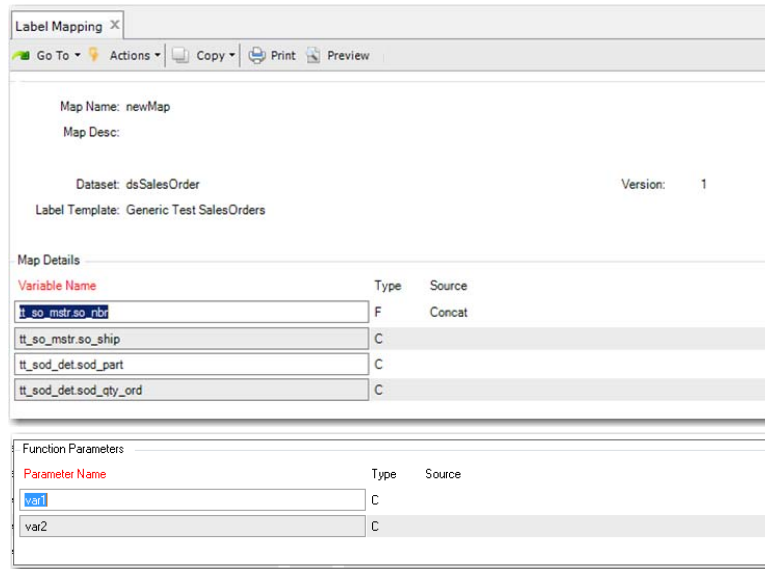
Label Mapping

Use Label Mapping (36.13.16.1.20) to map incoming data that does not match by name to other fields. When incoming data does not have a match, you can configure a label format to use the data in this program so that existing templates can still be used when matches do not exist.

Enter the map name, data set, and template in the header of the program, then press Go. The Map Details frame displays, showing the variables for the label template that need to be populated. You can select the variable name to provide a value for it.

When you use a function, a third frame displays the parameters to pass. They can be either type I (input) or C (constant).

Fig. 2.15
Label Mapping (36.13.16.1.20)



Variable Name	Type	Source
tt_so_mstr.so_nbt	F	Concat
tt_so_mstr.so_ship	C	
tt_sod_det.sod_part	C	
tt_sod_det.sod_qty_ord	C	

Parameter Name	Type	Source
var1	C	
var2	C	

Map Name. Indicate a unique identifier the system uses to find the map to use in label format details.

Map Desc. Optionally, enter a description of the map.

Version. Enter the version of the dataset.

Dataset. Enter a valid dataset name defined in the dataset maintenance screen.

Label Template. Enter a label template defined in the Label Template Maintenance.

Variable Name. Enter the variable name that requires mapping.

Type. Enter one of following:

F(unction): A valid function from the function maintenance screen.

C(onstant): A constant value – never changes.

I(nput): These are values coming in from the dataset. The system generates a list from the dataset from which you can select.

Configure Applications

Use Label Application Configuration (36.13.16.1.13) to configure applications. The program determines which applications are allowed to send requests to print labels.

Label Application Configuration has two frames. The first frame lets you set up the application and the code you provide for the application. It also determines if the application is active. The second frame lets you set up the default printing for specific contexts, such as function and user.

Fig. 2.16
Label Application Configuration (36.13.16.1.13)

Application ID. Indicate a unique ID to identify external applications that request labels to be printed.

Application Description. Optionally enter a description of the application ID.

Application Active. Indicate Yes to set the application as active. When No, the system does not process print requests from the application.

Dataset Name. Enter a valid and defined dataset name in Label Dataset Maintenance (36.13.16.1.12).

Context Key1. Enter a value to differentiate the data context routing. (The calling program sends it in an API.) The combination of the data in Context Key 1 and Context Key 2 informs the system which printer and format to use to print the data.

Context Key2: Enter a second value to differentiate the data context routing.

Format ID. Enter a valid and defined value in the format table. This is the generic label format name and is not specific to a printer type. The value defaults from the Printer ID field in this program.

Printer ID. Enter a valid and defined value in the printer table that is established by Label Printers (36.13.16.1.5). This sets the syntax the system uses to generate the actual barcode label.

Note The syntax and the format are the keys to the format detail table, which describes how the system generates the label; therefore, you only define the format and syntax here.

Set Up Contextual Routing

There are routing maintenance fields in the Application Configuration Maintenance program that let you specify a dataset, context keys, label format, and printer for a new application; however, if you need more advanced contextual routing, with exceptions specified for various print scenarios, you can use the following programs:

- “Label Content Routing Setup” on page 31
- “Label Content Routing Configuration” on page 33

The following topics describe the two programs, including fields and data to enter for the fields.

Wildcards

You can use wildcards when you set up data in these programs. You can specify full wildcard use using only an asterisk (*) to specify that the system search for all in the category (for example, specify an * for a site, and the system searches for all sites) or specify partial wildcard use by entering an asterisk before or after characters or letters, such as 10-*, where the system searches for values that begin with 10-.

Label Content Routing Setup

Use Label Content Routing Setup (36.13.16.1.15) to specify the dataset and purpose; for the specified dataset and purpose, provide the default printer and format.

You can then specify fields for a leveled hierarchy so that the system can determine which printer to use based on your setup when exceptions are needed. For example, you can specify that at the highest level, the system checks for Site100A during a PO receipt for printing specific labels at a specific printer.

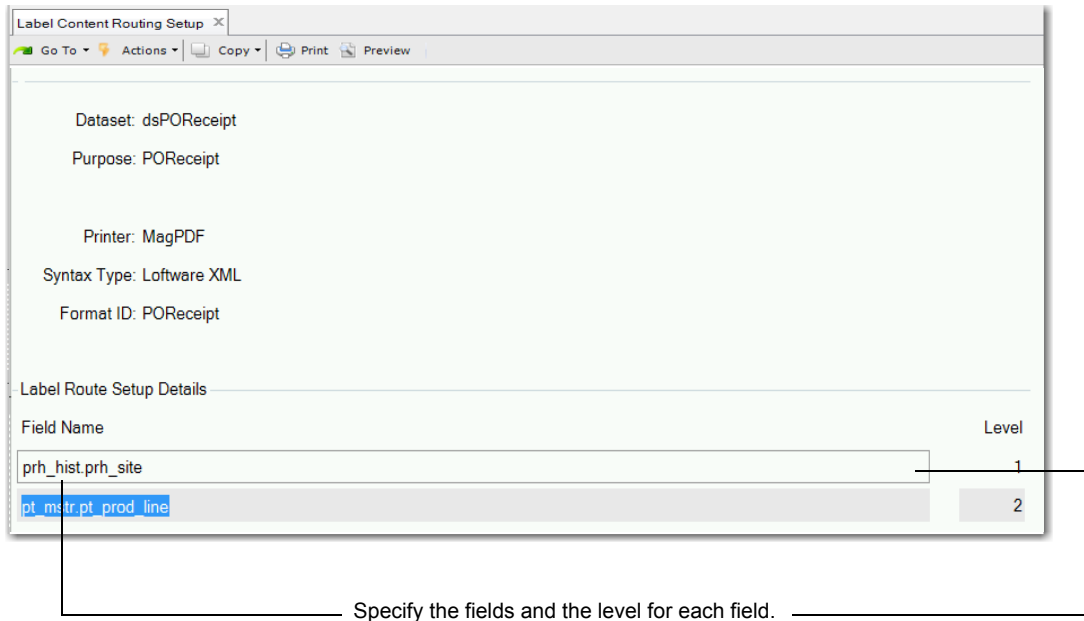
You can further refine the setup by specifying a certain vendor, for example, then using a certain label format and printer for that vendor. So, for example, when setting up a contextual routing to print labels for a PO receipt dataset, there may be several PO receipts that your company receives at site 130, and they all use a specific label format printed at a specific printer. However, when the vendor code is set for a specific vendor (Vendor123), then you can use a different label format and/or different printer.

Note Use Label Content Routing Configuration to complete the configuration for the specific-field-based printing setup using the specified levels; see “Label Content Routing Configuration” on page 33.

You can also delete the entire setup from the header frame without having to select each child component individually.

Figure 2.17 shows the field and level frame. You specify the fields and the level for each field. The system verifies that the value you specify in Label Content Routing Configuration for the level 1 field matches the data in the dataset. When it matches, it uses the label format and the printer specified at that level and for that value; otherwise, it uses the label format and printer at the header level. When a next level is specified, the system performs the same check as it did for level 1 to determine the correct format and printer for the next level.

Fig. 2.17
Label Content Routing Setup (36.13.16.1.15)



Dataset: dsPORReceipt
Purpose: PORceipt
Printer: MagPDF
Syntax Type: Loftware XML
Format ID: PORceipt

Label Route Setup Details

Field Name	Level
prh_hist.prh_site	1
pt_mstr.pt_prod_line	2

Specify the fields and the level for each field.

Dataset. Enter a valid dataset name defined in Label Dataset Maintenance (36.13.16.1.12).

Purpose. Enter the purpose of the label contextual routing setup. Although you can enter any text string, it is useful to enter a purpose that defines the reason for the contextual routing setup; for example, if you need labels for a PO receipt dataset, specify the purpose as PORceipt.

Printer. Enter a valid and defined value in the printer table that is established by Label Printer Maintenance (36.13.16.1.5). This sets the syntax the system uses to generate the actual barcode label.

Syntax Type. This field is view only. The system supplies the syntax type when you specify a printer. You define syntax types for printers in Label Printer Syntax Type Maint (36.13.16.1.2).

Format ID. Enter a valid and defined value in the format table. You define a format in Label Format Maintenance (36.13.16.1.9).

Field Name/Level. Enter a valid field name from the dataset and the level that the label engine checks to determine which label format to use for which printer.

Example You specify `prh_hist.prh_site` as the field name and the level as 1. The label engine checks the value in the field to see if there is a match. When there is, it uses the specific label format and printer. You also specify `pt_mstr.pt_prod_line` as field name and the level as 2. The engine checks the value in this second-level field only when the check at level 1 matches.

You can enter as many levels as required. There are no limits. You must set the printer and format for the various fields and levels, specify the value of the fields—for example, Site100A—and specify when to use each printer, format combination in Label Content Routing Configuration.

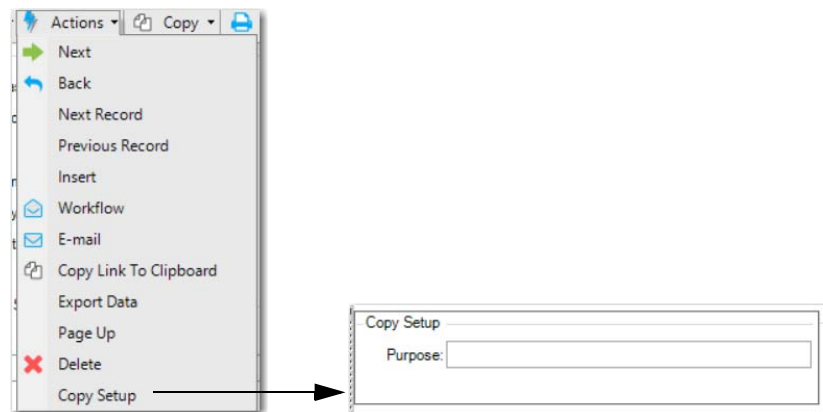
Copy Label Content Routing Setup

You can copy a label content routing setup definition to a new definition from the Actions drop-down menu.

To copy an existing label content routing setup, in Label Content Routing Setup, select the Copy Setup option from the Action drop-down menu. When copying a setup, the Copy Setup action is available once you reach the bottom frame and before you move on to individual detail records.

Once you select the Copy Setup action, a new frame appears prompting for the purpose of the new content routing setup definition that you are creating from the copied content (see Figure 2.18).

Fig. 2.18
Copy Setup



For setup copied content, the system verifies that:

- You have entered the purpose for the copied content.
- Your newly entered content routing setup definition does not already exist for another content routing setup definition.

Once the copy is complete, the system displays a confirmation message and prompts for the next record to enter.

Label Content Routing Configuration

Use Label Content Routing Configuration (36.13.16.1.16) to complete content routing by specifying values for the fields and levels you set up in Label Content Routing Setup (36.13.16.1.15). For example, if you specified in the setup program that the system checks for a site value first in a PO receipt, then a vendor, enter the value for the site and vendor here. You can specify a printer and format for each of the values at each level of the configuration.

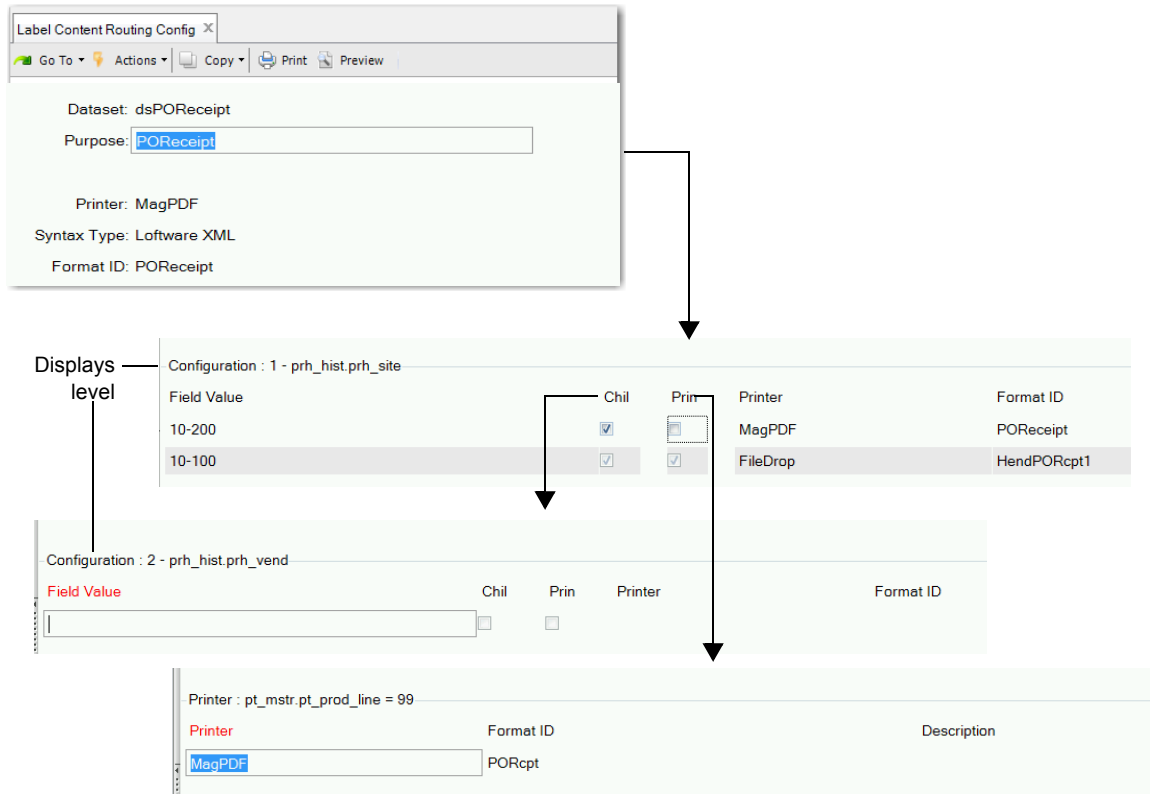
Once you enter the dataset and purpose, the system defaults the printer, format, and syntax from the setup program. When you press Enter, the system presents a second frame that lists the level fields from the setup program, starting with the level 1 configuration.

Use this frame to enter specific values for the fields; for example, enter 100A as the site value or Cust.123 for the vendor. You can set multiple values for the field, then specify that the system use different printers for that value.

Select Child to specify when you want to set additional values for the second-level fields in a setup; for example, if you want to specify that site 100A use printer 15 , but you set a second-level as a vendor, select Child, then enter the vendor values. Select Print to set a specific printer and format for the specific values.

Note When setting up second-level values, select Child, but deselect Printer; otherwise, the printer frame displays.

Fig. 2.19
Label Content Routing Configuration (36.13.16.1.16)



Dataset. Enter the dataset that you set up in Label Content Routing Setup (36.13.16.1.15).

Purpose. Enter the purpose of the label contextual routing setup.

Printer/Syntax Type. These fields are view only and defaults from Label Content Routing Setup.

Field Value. Enter a specific value for the field; for example, enter 100A for a specific site when you set the content routing to check for sites when printing specific labels. You set the fields and their levels in Label Content Routing Setup.

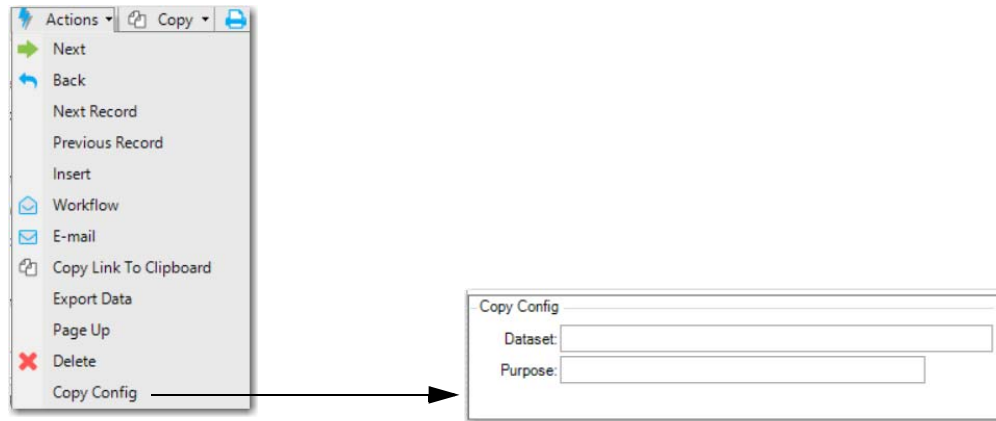
Printer. Specify a printer for each value that you enter for the fields.

Format. Specify a format for each value that you enter for the fields.

Copy Label Content Routing Configuration

To copy an existing label content routing configuration, in Label Content Routing Configuration, select the Copy Config option from the Action drop-down menu. When copying a configuration, the Copy Config action is available once you reach the bottom frame and before you move on to individual detail records. Once you select the Copy Config action, a new frame appears prompting for the dataset and purpose for the new content routing setup definition you are creating from the copied content routing configuration definition (see Figure 2.20).

Fig. 2.20
Copy Configuration



For configuration copied content, the system verifies that:

- You have entered the purpose and dataset name.
- The specified dataset name exists.
- A content routing setup record referencing the dataset name and purpose exists and contains at least one setup detail record.
- Your newly entered content routing configuration definition does not already exist for another content routing configuration definition.

Once the copy is complete, the system displays a confirmation message and prompts for the next record to enter.

Set Up Contextual Printer Routing

In the Label Content Routing Setup and Configuration programs, you set up the printer and the format (see “Label Content Routing Configuration” on page 33). But, when you have more complex printer routing requirements, you can set up the system to use the format first, then decide upon the printer to use. You can specify the format, the dataset, then you can choose to bypass any printer setup and use a specific printer based on the format and the dataset. To do this, you can use the following programs:

- “Label Printer Routing Setup” on page 36
- “Label Printer Routing Configuration” on page 38

The following topics describe the two programs, including fields and data to enter for the fields.

Note You can use wildcards when using these programs; see “Wildcards” on page 31.

Label Printer Routing Setup

Use Label Printer Routing Setup (36.13.16.1.17) to specify the dataset, the default printer, and the syntax type.

You can then specify fields for a leveled hierarchy so that the system can determine which printer to use based on your setup when you are ready to print for a specific purpose. For example, you can specify that at the highest level, the system checks for Site100A during a PO receipt for printing specific labels at a specific printer, then as a second level, the system checks by a specific production line, and at a third level, the system checks by receiver.

Figure 2.21 shows the variable name and level frame. You specify the variable names and the level for each. The system verifies that the value you specify for the level 1 variable name matches the data in the dataset. When it matches, the system uses the label format and the printer specified at that level; otherwise, it uses the label format and printer at the header level. When a next level is specified, the system performs the same check as it did for level 1 to determine the correct format and printer for the next level.

Fig. 2.21
Label Content Routing Setup (36.13.16.1.15)

The screenshot shows the 'Label Printer Routing Setup' window. It contains the following configuration details:

- Format ID: HENDPOReceipt
- Dataset: dsPOReceipt
- Printer: FileDrop1
- Syntax Type: File Drop

Below these details is a section titled 'Printer Route Setup Details' containing a table with two columns: 'Variable Name' and 'Level'.

Variable Name	Level
prn_hist_prn_site	1
pt_mstr_pt_prod_line	2

A callout box with the text 'Specify the fields and the level for each field.' points to the table.

Format ID. Enter a valid and defined value in the format table. You define a format in Label Printer Format Maintenance (36.13.16.1.9).

Dataset. Enter a valid dataset name defined in Label Dataset Maintenance (36.13.16.1.12).

Printer. Enter a valid and defined value in the printer table that is established by Label Printer Maintenance (36.13.16.1.5). This sets the syntax the system uses to generate the actual barcode label.

Syntax Type. This field is view only. The system supplies the syntax type when you specify a printer. You define syntax types for printers in Label Printer Syntax Type Maint (36.13.16.1.2).

Variable Name/Level. Enter a valid variable name from the dataset and the level that the label engine checks to determine which label format to use for which printer.

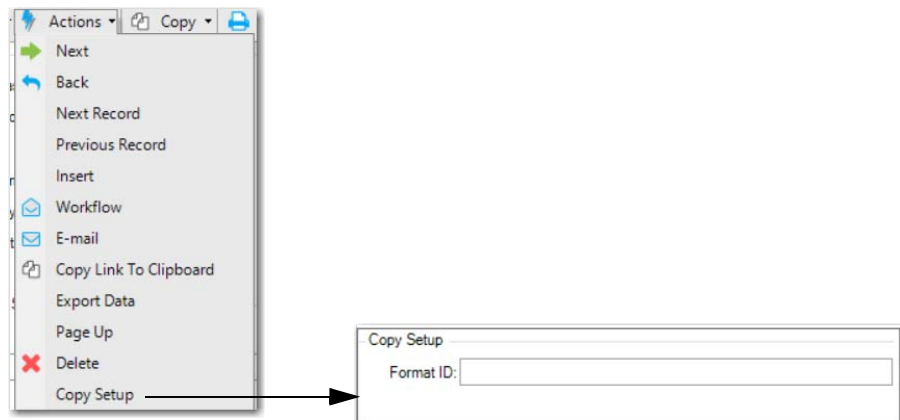
Example You specify `prh_hist.prh_site` as the variable name and the level as 1, the label engine checks the value in the field to see if there is a match and when there is, it uses the specific label format and printer. You also specify `pt_mstr.pt_prod_line` as variable name and the level as 2. The engine checks the value in this second-level variable name only when the check at level 1 matches.

You can enter as many levels as required. There are no limits. You must set the printer and format for the various variable names and levels, specify the value of the variable names—for example, Site100A—and specify when to use each printer, format combination in Label Content Routing Configuration.

Copy Label Printer Routing Setup

You can copy a label printer routing setup definition to a new definition. To do this, select Copy Setup from the Actions drop-down menu. The Copy Setup action is available once you reach the bottom frame and before you move to the individual detail records. Once you select Copy Setup, a new frame prompts you for the format ID with which to associate the new printer routing setup record.

Fig. 2.22
Copy Label Printer Routing Setup



For setup copied content, the system verifies that:

- You have entered the format ID for the copied content.
- An existing format is defined with this format ID.
- The format has a format detail with the syntax type matching the printer routing setup record being copied.
- Format detail is not already associated with another printer routing setup record.

Once the copy is complete, the system displays a confirmation message and prompts for the next setup record to enter.

Label Printer Routing Configuration

Use Label Printer Routing Configuration (36.13.16.1.18) to complete printer routing by specifying values that the system checks for the fields and levels you set up in Label Printer Routing Setup (36.13.16.1.17). This lets you set up printers to use when you configure Label Printing Services to use the format and dataset.

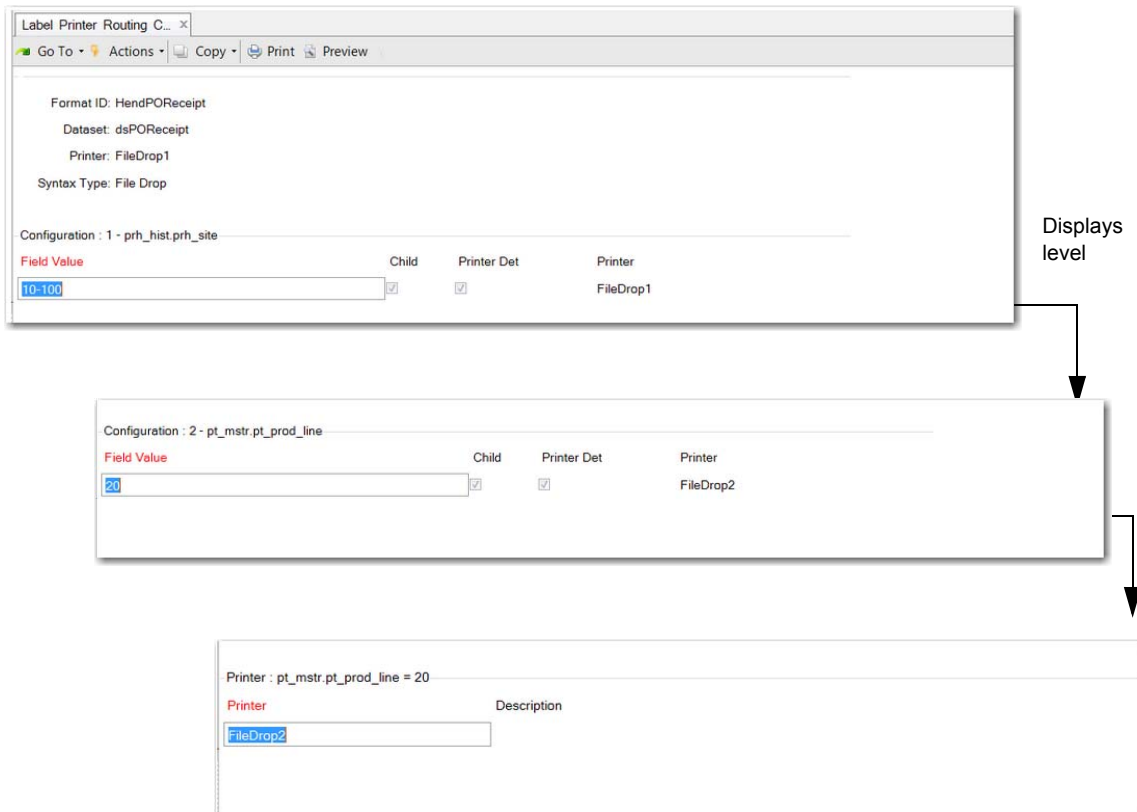
Once you enter the format ID and dataset, the system defaults the printer and syntax type from the setup program. When you press Enter, the system presents a second frame that lists the level fields from the setup program, starting with the level 1 configuration.

Use this frame to enter specific values for the fields. You can use wildcards when you enter the values. For example, enter 10-* as the site value or 100A for the production line. You can set multiple values for the field, then specify that the system use different printers for that value. You can use wildcards; see “Wildcards” on page 31.

Select Child to set additional values for the second-level fields in a setup; for example, if you want to specify that site 100A use printer 15, but you set a second-level as a production line, select Child, then enter the production line values. Select Print to set a specific printer and format for the values.

Note When setting up second-level values, select Child, but deselect Printer; otherwise, the printer frame displays.

Fig. 2.23
Label Printer Routing Configuration (36.13.16.1.18)



Format ID. Enter a valid and defined value in the format table. You define syntax types for printers in Label Printer Format Maintenance (36.13.16.1.9).

Dataset. Enter the dataset that you set up in Label Content Routing Setup (36.13.16.1.15).

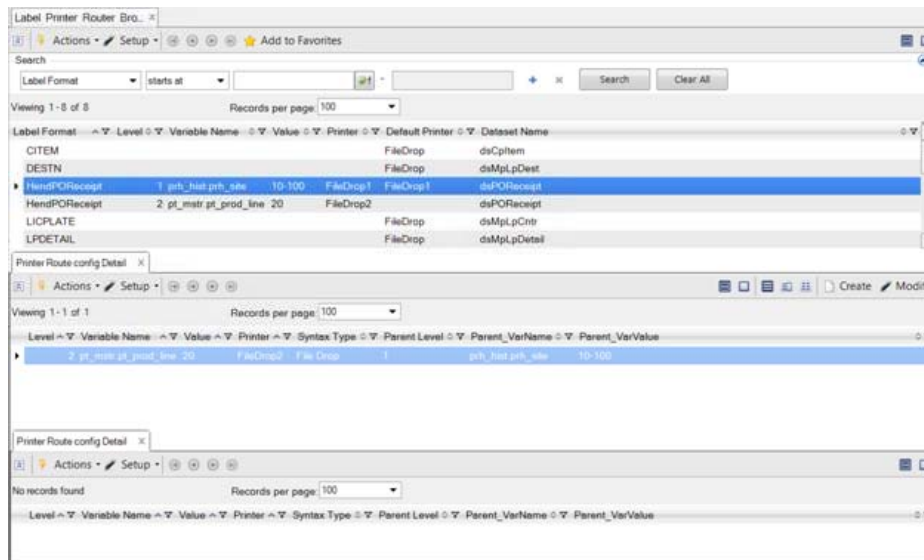
Printer/Syntax Type. These fields are view only and default from Label Content Routing Setup.

Field Value. Enter a specific value for the field; for example, enter 100A for a specific site when you set the content routing to check for sites when printing specific labels. You set the fields and their levels in Label Content Routing Setup.

Printer. Specify a printer for each value that you enter for the fields.

Format. Specify a format for each value that you enter for the fields.

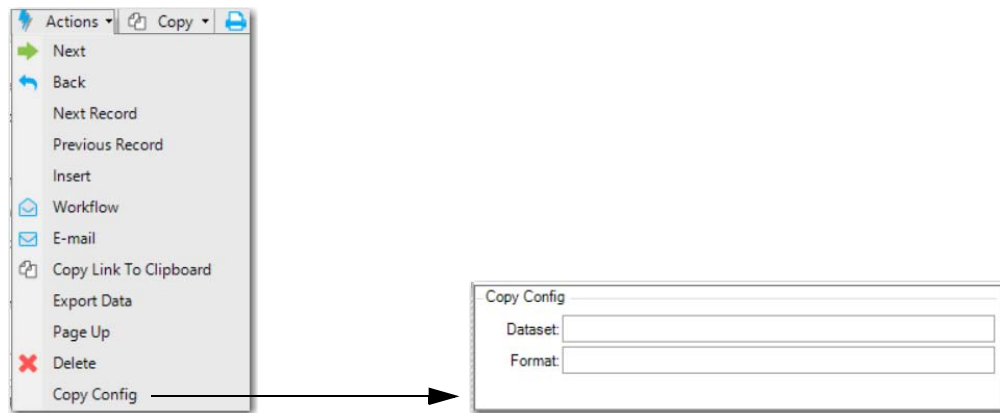
Fig. 2.24
Label Printer Router Browse



Copy Label Printer Routing Configuration

You can copy a label printer routing configuration definition to a new definition. To do this, select Copy Config from the Actions drop-down menu. When you select the action, a new frame prompts you for the dataset and format ID with which to associate the new printer routing configuration definition.

Fig. 2.25
Copy Configuration



The system verifies that:

- You have entered values for the dataset and format ID.
- The dataset and format ID entered are valid.
- An existing printer routing setup record exists that references the specified dataset name and format ID you entered
- The printer routing setup record found has at least one detail printer routing setup record
- A printer routing configuration record already exists with a printer routing setup record that matches the dataset name and format ID you specified.

Once the copy is complete, the system displays a confirmation message and prompts for the next setup record to enter.

Set Up Label Extraction

You can use the programs in the Label Extraction menu (36.13.16.3) to extract label data:

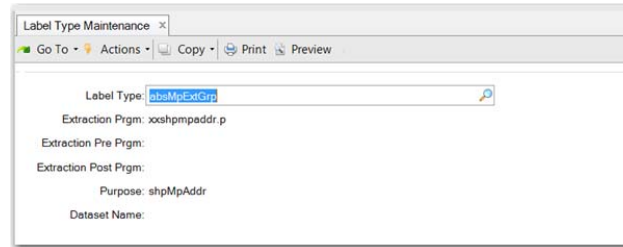
- Label Type Maintenance
- Label Group Maintenance
- Label Event Maintenance
- Label Event Processing
- Label Event Type Criteria Maintenance

In addition, this section explains the extraction process (see “Extraction Process” on page 45) and tells you how external modules can make calls to Label Printing Services using different protocols without considering the environment’s PROPATH (see “Make Calls without Considering PROPATH” on page 45).

Label Type Maintenance

Use Label Type Maintenance (36.13.16.3.1) to set up the label type, extraction programs, and determination for the middleware.

Fig. 2.26
Label Type Maintenance (36.13.16.3.1)



Label Type. Enter a unique ID for the label type. This field cannot be blank and cannot have spaces or special characters.

Extraction Prgm. Enter the name of the program that the system calls when the label type is being extracted. This field cannot be blank.

Extraction Pre Prgm. Enter the name of the program the system calls before the label type is extracted. This field can be blank, but if you enter a program it must exist in the PROPATH; refer to the Label Printing Services installation documentation for PROPATH information.

Extraction Post Prgm. Enter the name of the program the system calls after the label type is extracted. This field can be blank, but if you enter a program it must exist in the PROPATH; refer to the Label Printing Services installation documentation for PROPATH information.

Default Purpose. Enter the reason for which you pass information to the label printing engine. This field cannot be blank.

DataSet Name. Enter a valid dataset name.

Label Group Maintenance

Use Label Group Maintenance (36.13.16.3.4) to set up label grouping and determination for the middleware.

Use the first frame to define and maintain the label group definition. Once you define a group name, you can continue and provide a sequence of label types to be printed as a group.

Use the second frame to enter a sequence. When a sequence is available then you can update it; otherwise, create a new one. You specify the label type and purpose, and indicate whether the sequence has a child. The sequence is the sequence in which the label type attempts to print. This is the order in which the labels print within the defined level.

You can also have additions and deletions; you delete using F5 from the scrolling window.

Fig. 2.27
Label Group Maintenance (36.13.16.3.4)

Seq	Label Types	Purpose	Child
1	ebsMpExtGrp	shpMpAddr	<input checked="" type="checkbox"/>

Group ID. Enter a unique ID for the label type. This field cannot be blank.

Description. Provide a description for the label group.

Sequence. The sequence in which the label type attempts to print.

Label Type. Add a label type to be printed.

Purpose. Provide a reason for the label to be printed. This will be used to determine format and printer in label content routing (see “Label Content Routing Setup” on page 31).

Child. Select if you want to define child label types for the current sequence.

Label Event Maintenance

Use Label Event Maintenance (36.13.16.3.13) to set up events that other QAD programs can use to trigger label print from the Label Printing Services.

In the first frame, enter information for the event, such as the event ID, table name, field name, additional tables, and the join phrase. In the Label Event Field Details frame, define the fields used to determine when data extraction should occur and labels should be generated.

Fig. 2.28
Label Event Maintenance (36.13.16.3.13)

Sequence	Field Name
1	cp_mstr.cp_cust

Event ID. Enter the event name used by the calling program—for example, QAD Data Collection—when triggering a label event.

Table Name. Enter the name of the table from which key data is passed.

Field Name. Enter the name of the field in the table for which the key value is going to be picked.

Additional Tables. Enter a comma-separated list of tables used in the query.

Join Phrase. Enter a comma-separated list of joins associated with the tables defined in the Additional Tables field.

Description. Enter a description of the event.

Field Name. Enter the field name to determine (rules based) when the data extraction occurs. There can be many records for a given label event. Each record created must be a field in the table listed in the first frame of the program.

Sequence. This is the order of the fields in the label group setup. The sequence lets you create a list of fields and match the values being passed from the main and additional tables listed in Label Event Maintenance against the configuration in Label Event Type Criteria Maintenance.

Label Event Processing

Use Label Event Processing (36.13.16.3.14) to tie the event and the label group together. There can be multiple label groups per event.

Fig. 2.29

Label Event Processing (36.13.16.3.14)

Event ID. Enter an event ID defined in Label Event Maintenance. This field cannot be blank.

Group ID/Description. Enter a valid group ID as set in Label Group Maintenance.

Label Event Type Criteria Maintenance

Use Label Event Type Criteria Maintenance (36.13.16.3.15) to complete the extraction setup. Use the following procedure:

- 1 Specify an Event ID defined in Label Event Maintenance.
- 2 Select Browse Collection.

The system displays a list of all label types from all label groups linked to this label event in Label Event Processing.

- 3 Specify a label type; then select Next.

The system displays a frame that lets you specify sets of accepted values for the fields you selected in Label Event Maintenance.

Note You can enter blank values in fields. When you do, the system saves the entry and displays the blank value as double quotation marks (“”). Entering either double or single (“”) empty quotation marks is equivalent to a blank field.

- 4 Select a line and select Next to modify the values.

Note You can use wildcards; see “Wildcards” on page 31.

- 5 In the same frame, specify a default label format and number of copies.

The system matches the values passed from the tables listed in Label Event Maintenance against the values you set up in Label Event Type Criteria Maint. If there is no match, no further processing will be done, thus no labels will be generated.

Fig. 2.30
Label Event Type Criteria Maint (36.13.16.3.15)

The figure illustrates the workflow in the 'Label Event Type Criteria Maint' application. It consists of three sequential screenshots connected by arrows:

- Top Screenshot:** Shows the main form with 'Event ID: ItemLabelEvent' and 'Browse Collection' checked. Under 'Label Type', 'ItemLabelType' is selected in a dropdown menu, and the 'Config' checkbox is checked.
- Middle Screenshot:** Shows the 'Values' section with 'ItemLabelType: pt_mstr;pt_abc' entered. Below it, a table shows 'Format ID' and 'Copies' with values '1'.
- Bottom Screenshot:** Shows a detailed 'Label Criteria' table for 'ItemLabelType: pt_mstr;pt_abc'. The table has columns for 'Field' and 'Value'.

Field	Value
pt_mstr;pt_abc	*
Format ID	
Copies	1

Event ID. Enter the event ID defined in Label Event Maintenance.

Browse Collection. Select to use display mode.

Label Type. The system displays a list of all label types from all label groups linked to this label event in Label Event Processing.

Values. Add or remove a line to allow a particular set of values.

Config. Selected when a label event type criteria configuration has been defined for a particular label type.

Field. A field selected in Label Event Maintenance or Format ID and the number of copies.

Value. The allowed value for the field or the default Format ID and number of copies.

Extraction Process

The data extraction process receives the necessary key values to start the extraction and call Label Printing Services. The process ensures that the following are correct:

- Looping

Label Printing Services can only produce a label for the outermost loop of the dataset; therefore, the data extraction process ensures that the outermost looping level is the same as the number of labels required—for example, the packing level should be the outermost when you produce pack labels and should not loop on the shipper or the items.

- Content

The data extraction process ensures that the naming conventions are considered when sending data to the label printing system. Label Printing Services uses data templates with naming conventions that match the dataset. The data extraction process produces a dataset definition that can be imported into the Label Printing Services.

API and Processing

You can create a record in the label event workfile (LBEVNT_WKFL) as the trigger point.

To create the label event workfile (LBEVNT_WKFL) record:

- 1 Use the `createLPSWorkFileRecord` procedure from the `lblstdxr.p` program.
- 2 All calls are made to Label Printing Services by setting the `c-application-mode` variable to `API` in the calling program. Set the variable to `API` before calling Label Printing Services. After the call, revert the `API` value back to the original `c-application-mode` value in the calling program.
- 3 Use the `deleteLPSWorkFileRecord` procedure from `lblstdxr.p` program to delete the label event workfile (LBEVNT_WKFL) record.

Important QAD ICT is not required to call Label Printing Services using the workfile since the Label Printing Services trigger file is built into the Label Printing Services caller program.

The data extraction process runs according to the parent-child relationship defined in the Label Group Type (`lbGrpTypes`) table. The system calls the program defined in the label type extraction.

For the subscribing software, the subscriber starts the determination process, using the setup you specify in the Label Extraction Configuration programs.

Make Calls without Considering PROPATH

Version 1.1.30 and higher includes the `lbcallsps.p` program, which lets external modules such as QAD EAM, QAD Warehousing, and so on, make calls to Label Printing Services using different protocols. The program lets the programs make direct calls without considering the environment's `PROPATH` where Label Printing Services is installed. The program includes a procedure for calls coming using different protocols.

The `lbcallsps.p` contains a `PrintLabelViaAppServer` procedure that can be used when calling Label Printing Services through the `AppServer`. The contents of the program are as follows:

```

/* lbcalllps.p - Wrapper program to call any LPS Program                               */
/* Copyright 1986 QAD Inc. All rights reserved.                                       */
/* $Id::                                             $: */
/*                                                                                       */
/* Program for mng via different protocols (AppServer, Webservice, etc                 */
/*                                                                                       */
/* *****                                                                                       */
/* All patch markers and commented out code have been removed from the source */
/* code below. For all future modifications to this file, any code which is      */
/* no longer required should be deleted and no in-line patch markers should     */
/* be added. The ECO marker should only be included in the Revision History.    */
/* *****                                                                                       */
{mfsubdirs.i}
{{&US_BBI}mfdeclre.i }
{{&US_PX}pxttmsg.i}
/* Call via AppServer */
procedure PrintLabelViaAppServer:
  define input  parameter pcPurpose as  character no-undo.
  define input  parameter dataset-handle  phDataSet bind.
  define output parameter table for temp_err_msg.

...

end procedure.

```

Import/Export Label Setup Data

You can import or export label, printer, and routing setup data from Label Printing Services from or to the environment of your choice using the programs in the Label Extraction Configuration Menu (36.13.16.4):

- Label Config Import/Export
- Label Extraction Import/Export

Label Config Import/Export

Use Label Config Import/Export (36.13.16.4.1) to select the label, printer configuration, or label format setup data to export from or import to a specific environment. You can choose the setup data to be exported or imported by specifying all labels, printer configuration only, or format configuration only. You can also choose to select all and overwrite setup data in the environment.

Fig. 2.31

Label Config Import/Export (36.13.16.4.1)

```

Import/Export: Export
Options: (X) All Label Information
         ( ) Printer Configuration Only
         ( ) Format Configuration Only

Directory: /qad/local/sandbox/user/vvk/01/ee2012 1/build/work/repo
File Mask: *
Select All: No
Overwrite: No

Output:

```

Import/Export. Specify whether the function imports or exports setup data.

Options. Specify the setup data for the function:

All label information: The system imports or exports all label setup data to or from the folder you specify in the Directory field.

Printer Configuration Only: The system imports or exports only printer configuration setup data to or from the folder you specify in the Directory field.

Format Configuration Only: The system imports or exports only label format setup data to or from the folder you specify in the Directory field.

Directory. The directory to which the system exports setup data or from which you import setup data.

File Mask. Any masks that the system uses when selecting data; for example, an asterisk (*) to use as a wildcard.

Select All. Indicate whether to select all.

Overwrite. Indicate whether to overwrite data in the directory you specify when importing or exporting.

Label Extraction Import/Export

Use Label Extraction Import/Export (36.13.16.4.2) to export point-of-print and label-data extraction setup. Specify the operation, then specify the directory, the source code directory, and whether to use a mask, select all, or overwrite data.

Fig. 2.32
Label Extraction Import/Export (36.13.16.4.2)

```

Import/Export: Export
Directory: /qad/local/sandbox/user/vvk/01/ee2012 1/build/work/repo
Source Code Directory:
File Mask: *
Select All: No
Overwrite: No
Output:
  
```

Import/Export. Specify whether the function imports or exports setup data.

Directory. The directory from which the system exports setup data or to which you import setup data.

Source Code Directory. Specify the source code directory for the operation you specify.

File Mask. Any masks that the system uses when selecting data; for example, an asterisk (*) to use as a wildcard.

Select All. Specify whether to select all.

Overwrite. Specify Yes to overwrite data in the directory you specify when exporting or overwrite existing Label Printing Services point-of-print and label-data extraction setups.

Delete Stranded Work Flows

Use Remove Stranded Label Work File Records (36.13.16.4.14) to delete stranded label event work flow records (lbevntwkfl records). The system creates these records when processing labels and typically deletes them once the label is processed; however, occasionally, they are not successfully removed from the system.

When you run this utility, the system prompts you to enter a date. The system deletes the work flow records with a date prior to this date. The default is today's date.

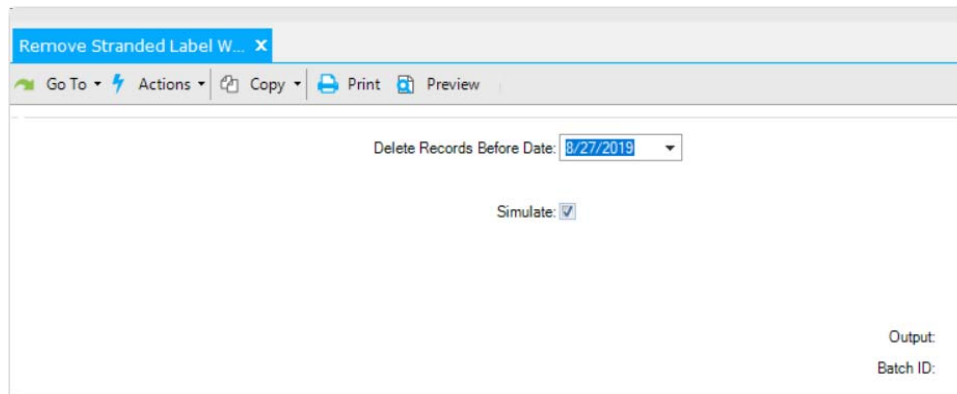
The utility also prompts you to indicate whether to run the utility in simulation mode. When you specify Yes for simulation mode, the system creates an archive file to report the records it *intends* to delete once you actually run the utility. The system defaults to run in simulation mode. The reported data in the archive file displays the word `simulation` at the start of each line in the before each record, and both `simulation` and `archive` are in the file name along with the date.

When you run the utility to actually delete the records by setting Simulate to No, the system creates an archive file in your working directory that reports the date and time the deletion executes and the number of records deleted. The system displays the name of the archive file the system creates.

You can also run this utility in batch mode.

Fig. 2.33

Remove Stranded Label Work File Records (36.13.16.4.14)



Set Up Labels for QAD Serialization

When you use QAD Serialization to print labels with serial IDs, LPS uses configuration data to determine the correct format and printer for Serialization labels. To do this, LPS uses a label event workfile and label content routing setup and configuration data.

Note Prior to the LPS version 3.3 release, QAD Serialization transactions used item/pack label setup instead of the LPS routing methods.

To set up label data in QAD Serialization, use Pack Label Format Maintenance (13.14.13) to define a label format code for a specific packaging type or Item Label Format Maintenance (13.14.16) to define label format codes for specific inventory items. For more information, refer to the *QAD Serialization User Guide*.

Label Event Workfile

To print serial labels using configuration, based on an Automation Solutions: Data Collection transaction or a custom program as the calling program, you create a label event workfile with the following data:

- An event ID as SerialPrint
- An OID (object ID) value of the serial master record

When the calling program makes a print request to LPS, the calling program passes event and label data to LPS to create a record in the labelEventWkfl table. The table holds the information until the serial labels print.

The LPS `lbsubscr.p` program is triggered automatically when the labelEventWkfl record is created and uses event and event processing information to process all label groups and group types (label types belonging to the group) associated with the event.

The system determines the label data to extract for a label print and processes the information as defined in Label Event Type Criteria for the serial label types.

For serial labels, LPS configuration includes content routing data for Serialization transactions. You can print serial labels for any Serialization transaction by passing the Serial OID and Event ID (SerialPrint). Refer to “Serial Label” on page 80 in chapter 4 to see the extraction program, template, format, and dataset for a serial label.

Data Collection Transactions

For labels that are printed through Automation Solutions: Data Collection (DC) transactions, the Data Collection user information includes DC transactions that print labels. Refer to the latest *QAD Automation Solutions: Data Collection Transactions User Guide* in the QAD Document Library for information on transactions.



Processing with Label Printing

This chapter discusses the following topics:

Processing Overview 52

Introduces processing with label printing.

View Label Print Status and Reprint Labels 53

Tells you how to use the Label Print Status Collection as a one-stop program to view label printing data, search for specific labels, or reprint labels.

View Labels 56

Tells you how to use various Label Printing Services programs to view or find labels.

View Router Configuration Data 59

Tells you how to create and modify label printer routing configuration records using the Label Printer Router Collection.

Print Labels Manually 60

Describes Label Printing Services programs that let you print labels manually.

Automatically Start and Release Requests from Queues 64

Tells you how to use Auto Release Startup to have the system automatically start and release label print requests.

Making Calls to the Label Printing Services 65

Provides information to help you make calls to Label Printing Services.

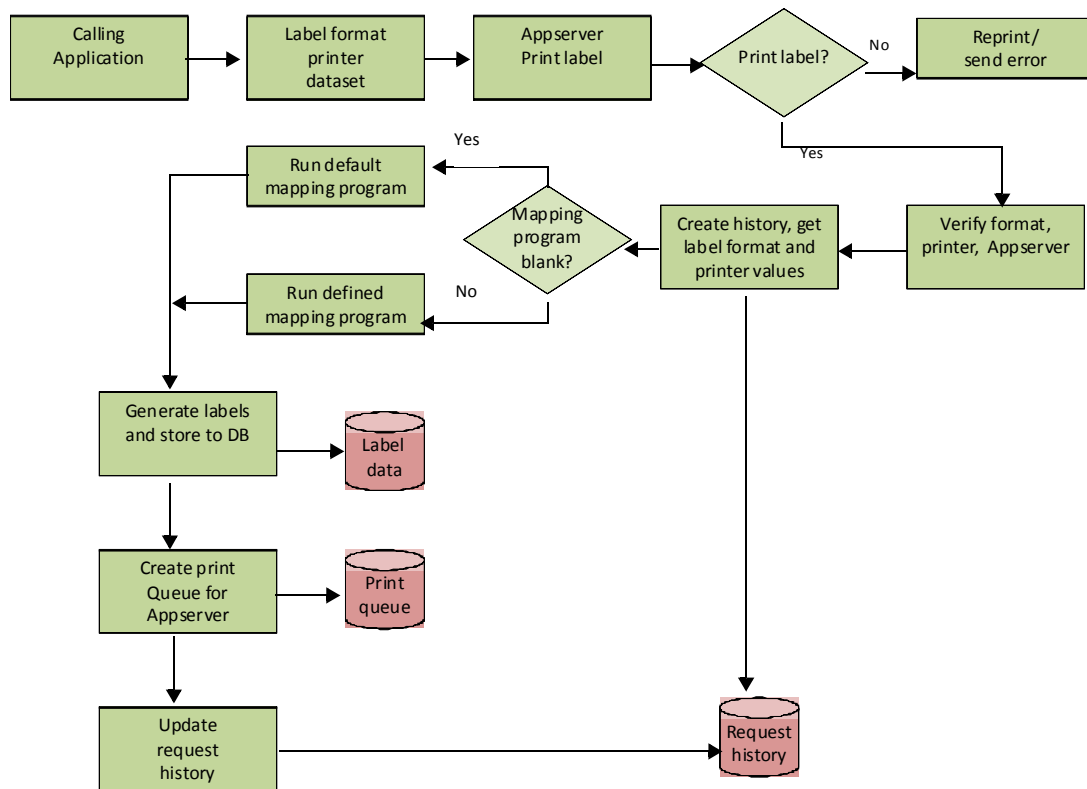
Processing Overview

Once you set up Label Printing Services with programs in the Label Printing Configuration menu (36.13.16.1), you can use the programs in Label Printing Operations (36.13.16.2) to:

- View and maintain information regarding the request and the data that supports it.
- Monitor labels queued to be printed.
- View the actual representation of a label.
- Manually print queued labels.
- Start the AppServer for printing queued labels for a polling frequency.
- Generate labels in an ad hoc manner.
- Release queued labels based on user input.

The following graphic presents an overview of the label-printing process. For setup information, see “Label Print Setup” on page 11.

Fig. 3.1
Processing Overview



You use the programs in the Label Printing Operations (36.13.16.2) menu to process orders with printed labels.

- Label Request History (36.13.16.2.1)
- Label Queue Monitor (36.13.16.2.3)
- Label Maintenance (36.13.16.2.5)
- Label Print Status Browse (36.13.16.2.6)

- Label Reprint Maintenance (36.13.16.2.7)
- Label Manual Release (36.13.16.2.13)
- Label Auto Release Startup (36.13.16.2.16)
- Ad Hoc Label Generation (36.13.16.2.19)
- Manual Label Reprint (36.13.16.2.14)

The descriptions include program and field descriptions, examples, and screen captures.

View Label Print Status and Reprint Labels

The following topics discuss ways to view label print status as well as reprint labels through a Label Printing Services browse collection.

Label Print Status Collection

You can use the Label Print Status Collection as a one-stop program to view label printing data, search for specific labels, or reprint labels.

The browse collection lets you select a print request from the top-level browse, then drill down into specific request data in supporting browses. Both label print and reprint requests are visible. You can view the:

- Status of labels generated by printer, date, and time based on business needs
- Status of label requests, regardless of whether you choose to store the label data
- Name of the QAD module requesting label print; for example, QAD EAM or QAD Serialization
- Application data that generated the label
- Content of the label master particular to the request ID

You can use Label Print Status Collection to access Label Maintenance; see “Maintain Labels from Label Print Status Collection” on page 54.

You can also use Label Print Status Collection to reprint labels; see “Reprint Labels from Label Print Status Collection” on page 55.

You can also view the label request history; see “Label Request History” on page 56.

Fig. 3.2
Label Print Status Collection

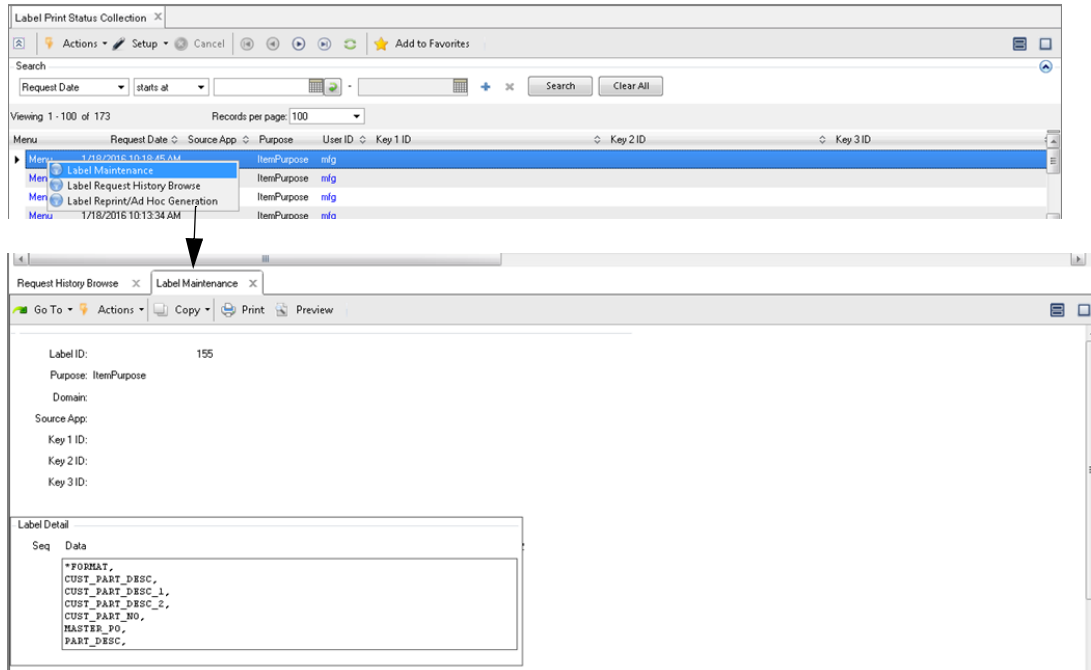
The screenshot displays four stacked windows from the QAD Label Print Status Collection interface:

- Label Print Status Browse:** Shows a search bar and a table with columns: Menu, Request Date, Source App, Purpose, User ID, Key 1 ID, Key 2 ID, Key 3 ID, Label Format, and External Device Printer. The table contains three rows of data for menu items printed on 9/16/2015.
- Request History Browse:** Shows a table with columns: Source App, Request ID, Copies, Request Date, Status, Request Type, and Request ID. It displays one record for request ID 319.
- Request History Response Browse:** Shows a table with columns: Request ID, Request Date, and Status. It displays one record for request ID 320.
- Request History Detail Browse:** Shows a message sequence table with columns: Request ID, Message Sequence, Message Number, and Message. It displays "No records found".

Maintain Labels from Label Print Status Collection

You can access Label Maintenance from the Label Print Status Collection by right-clicking the label of interest, then selecting Label Maintenance from the drop-down menu that displays. This lets you redefine the label without leaving the collection.

Fig. 3.3
Label Print Status Collection, Label Maintenance



Reprint Labels from Label Print Status Collection

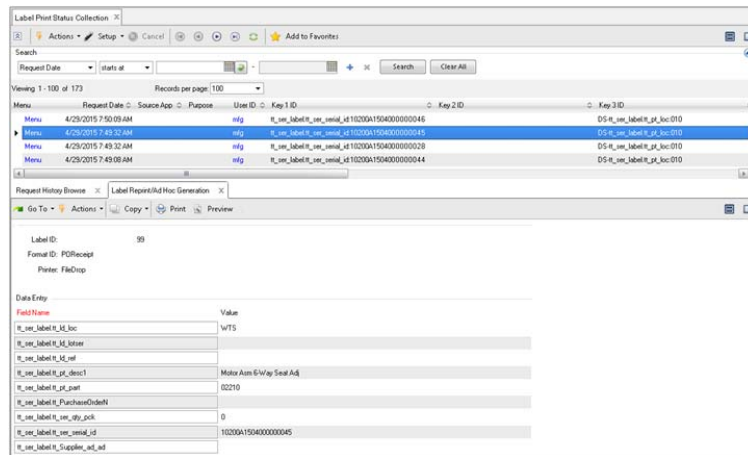
To reprint a label, right-click the Menu column in the row with the label of interest. You can choose to be redirected to Label Reprint/Ad Hoc Label Generation. The system preselects the label or request in the redirected program that you selected in the Label Print Status.

When you access Label Reprint/Ad Hoc Label Generation from the Label Print Status Collection, you can reprint or create new labels based on existing ones.

You can enter up to 500 characters in the Value field in a separate frame that displays when you enter the field. When you keep original label values, the system reprints them. When you modify values, the system creates a brand new label.

For field information, see “Ad Hoc Label Generation” on page 63. The following graphic depicts the program when you access it from within the Label Print Status Collection.

Fig. 3.4
Label Print Status Collection, Label Reprint/Ad Hoc Label Generation



View Labels

Use the following programs to help you view or find labels:

- Label Request History
- Label Queue Monitor
- Label Maintenance
- Label Reprint Maintenance

Label Request History

Use Label Request History (36.13.16.2.1) to view data about the request for a label print or reprint. Label Request History displays source data in readable form, making it easy to monitor and analyze label data. Setting the Display field to Table displays the source data in a field/value combination, making it easy to read. The field defaults to table data.

Fig. 3.5
Label Request History (36.13.16.2.1)

The screenshot shows the 'Label Request History' application window. The main view displays details for a specific request:

- Request ID: 63
- Src App Req ID: 62
- Success:
- Type: PRINT
- Request Date: 3/31/2015 12:02:49.605
- Copies: 1
- Status: PRINTED
- Display: TABLE
- Printer: No_Print
- Format ID: POLABEL

Below the main view is an 'Original Message' section containing the following text:

```

tt-P0Type.Device : >>><<<
tt-P0Type.LabelFormatID : >>><<<
tt-P0Type.LabelQty : >>>1<<<
tt-P0Type.LabelUserId : >>>mfq<<<
tt-P0Type.oid_pt_mstr : >>>201101060004079617.0009<<<
tt-P0Type.pt_chr01 : >>><<<
tt-P0Type.pt_chr02 : >>><<<
tt-P0Type.pt_chr03 : >>><<<
  
```

At the bottom right is a 'Processing History' table:

Requ	Status
07/28/2011 08:03:46.955	GENERATED
07/28/2011 08:03:55.075	PRINTED
07/28/2011 08:03:55.079	PRINTED

An annotation with an arrow points to the 'Success' checkbox, stating: "Set to Yes to view print request success."

Request ID. Indicate the request ID to view.

Src App Request ID. Indicate the request ID from the source application that made the request.

Success. Indicate Yes to view a list of successfully processed print requests.

Display. Select Table to have the system displays source data in a field/value combination, making it easy to read. Select XSD to have the system display source data as an XML file. This field defaults to table data.

Type. The system indicates the type of request; for example, print or reprint.

Request Date. The system displays the request date.

Copies. The system indicates the number of copies printed.

Status. The system indicates the status.

Printer. The system indicates the printer.

Format ID. The system indicates the format identification, as defined in Label Format Maintenance (36.13.16.1.9).

Label Queue Monitor

Use Label Queue Monitor (36.13.16.2.3) to view labels the system queues to print. You do not enter data in this program, but you can delete a record from a queue. The system displays the request ID, queue, and the request date.

Fig. 3.6
Label Queue Monitor (36.13.16.2.3)

Request ID	Queue	Create Date
5	prntque001	04/12/2012 02:59:05.959
6	prntque001	04/12/2012 03:15:27.114
7	prntque001	04/17/2012 23:42:49.740
8		04/18/2012 01:28:12.404
9	Error	04/18/2012 01:30:19.763
10	prntque001	04/18/2012 01:44:02.703
11	Error	04/18/2012 02:21:02.429
12	Error	04/18/2012 02:21:59.486
13	Error	04/18/2012 02:28:14.260
14	Error	04/18/2012 05:30:45.858
15	Error	04/19/2012 01:33:11.255
16	prntque001	04/19/2012 04:29:11.921
17	prntque001	04/19/2012 04:35:33.289
18	prntque001	04/19/2012 06:19:59.933
19	prntque001	04/19/2012 08:19:48.893
20	Error	04/19/2012 17:24:22.845

Label Maintenance

Use Label Maintenance (36.13.16.2.5) to view the label as it was generated.

After you enter header data and press Go, the system displays the status, request ID and format ID, create date and print date, and the file name where the system stores the request history in display-only fields. The system then displays the actual generated data; however, you cannot update the data.

Fig. 3.7
Label Maintenance (36.13.16.2.5)

Label Maintenance

Label ID: 2

Purpose:

Domain:

Source App:

Key 1 ID:

Key 2 ID:

Key 3 ID:

Status: PRINTED

Copies: 0

Create Date: 4/11/2012 20:23:56.317

Print Date: 4/11/2012 20:23:56.327

Request ID: 2

Format ID: PNDFormat

Printer: PNDTestFile

File Name: /users/pnd/vvk/PND-2012041184236321-2.PND

Label Detail

Seq	Data
1	<pre> <variable name="PackCode"></variable> <variable name="Qty"></variable> <variable name="SerialNumber"></variable> <variable name="Address"></variable> </label> </labels> </pre>

Label ID. Indicate the label ID whose data you want to view.

Purpose. Indicate the purpose of the label. For example, indicate when this is a shipment label.

Domain. Indicate the domain where the label data resides.

Source App. Indicate the ID of the source application that requested the label to print.

Key 1 ID - Key 3 ID. Because certain data may have more than one unique identifier, use the three keys to uniquely identify the label for searching purposes. These are values that are assigned to these fields by using the Label Format Maintenance token values. The source application, domain, and purpose also can help identify the label.

Label Reprint Maintenance

Use Label Reprint Maintenance (36.13.16.2.7) to maintain or search for reprinted labels. You can specify the request ID of the original request as well as the label ID, status, and file name of the stored label data. The system supplies today's date as the reprint date. You can use Label Request History (36.13.16.2.1) to obtain data for the fields in this reprint program.

Fig. 3.8
Label Reprint Maintenance (36.13.16.2.7)

View Router Configuration Data

You can view label printer routing configuration records using Label Format Router Browse. Note that you must define the content (format) routing to view data; however, viewing printer routing is optional.

Fig. 3.9
Label Format Router Browse

The screenshot shows three stacked windows from the QAD Label Format Router Browse application. The top window displays a list of datasets with columns for Dataset Name, Purpose, Level, Variable Name, Value, Label Format, and Device ID. The middle window shows the details for the selected dataset 'dsPORReceipt'. The bottom window shows the details for another dataset 'dsPORReceipt'.

Dataset Name	Purpose	Level	Variable Name	Value	Label Format	Device ID
dsMplMstr	MplMstr	2	tr-MplMstr_abs_shipto			
dsPORReceipt	PORReceipt	1	prh_hist_prh_site	10-200	HendPORReceipt	HendPORReceipt
dsPORReceipt	PORReceipt	1	prh_hist_prh_site	10-300	HendPORReceipt	HendPORReceipt
dsPORReceipt	PORReceipt	2	prh_hist_prh_nbr	1234	HendPORReceipt	HendPORReceipt
dsPORReceipt	PORReceipt	2	prh_hist_prh_nbr	234	HendPORReceipt	HendPORReceipt
dsPORReceipt	PORReceipt	2	prh_hist_prh_nbr	12345	HendPORReceipt	HendPORReceipt

Dataset Name	Purpose	Level	Variable Name	Value	Label Format	Device ID
dsPORReceipt	PORReceipt	2	prh_hist_prh_nbr	12345	HendPORReceipt	HendPORReceipt

Dataset Name	Purpose	Level	Variable Name	Value	Label Format	Device ID
dsPORReceipt	PORReceipt	1	pr_hist_prd_line	120	HendPORReceipt	HendPORReceipt

Print Labels Manually

You can use any of the following programs to print labels manually:

- Label Manual Release
- Manual Label Reprint
- Ad Hoc Label Generation
- Label Reprint/Ad hoc Generation

Label Manual Release

Use Label Manual Release (36.13.16.2.13) to print queued labels manually. You can use Label Queue Monitor (36.13.16.2.3) to view labels the system queues to print. This is useful when there are many labels in the queue and you need to print a label immediately.

Enter data in the header, which is similar to QAD EE report programs in which you specify a range, then press Go. The system displays the labels in the print queue. In the Sel column, change No to Yes to print the label, then press Go.

Fig. 3.10
Label Manual Release (36.13.16.2.13)

Change the Sel fields from No to Yes to print manually.

Labels to Release	Format			
<input type="checkbox"/> No	PNDFormat			
<input type="checkbox"/> no	PNDFormat			
<input type="checkbox"/> no	PNDFormat	12	4/18/2012	
<input type="checkbox"/> no	PNDFormat	13	4/18/2012	
<input type="checkbox"/> no	PNDFormat	14	4/18/2012	
<input type="checkbox"/> no	PNDFormat	15	4/18/2012	
<input type="checkbox"/> no	PNDFormat	16	4/19/2012	
<input type="checkbox"/> no	PNDFormat	39	4/19/2012	
<input type="checkbox"/> no	PNDFormat	67	4/23/2012	
<input type="checkbox"/> no	PNDFormat	68	4/23/2012	
<input type="checkbox"/> no	PNDFormat	120	5/2/2012	
<input type="checkbox"/> no	PNDFormat	121	5/2/2012	
<input type="checkbox"/> no	PNDFormat	122	5/4/2012	
<input type="checkbox"/> no	PNDFormat	123	5/4/2012	

Request Type. Indicate the type of request, either print or reprint.

Print Queue. Indicate the print queue number. Use Print Queue Monitor to find the queue ID.

From/To Date. Specify a range of print request dates.

From/To Request ID. Specify a range of print request IDs.

From/To Format ID. Specify a range of valid format IDs as defined in Label Format Maintenance (36.13.16.1.9). This is the request identifier sent from the calling application as a parameter in the actual call to the label print engine.

From/To App ID. Specify a range of source application IDs.

From/To From Request ID. Specify a range of print request IDs. Use the lookup browse to find the IDs. This is the internal request ID. It is a Progress OpenEdge database sequence that is updated on each call to the label print engine.

From/To Label. Specify a range of labels.

From/To App. Specify a range of applications. This is the name of the source application sent from the calling programs as a parameter.

From/To Domain. Specify a range of domains from which to print.

From/To Purpose. Specify a range of purposes to print. This is user-defined by the calling application. It can be blank or populated with the data from the caller depending on whether you send it using an API.

From/To Label Key 1- Label Key 3. Specify a range of key values.

Manual Label Reprint

Use Manual Label Reprint (36.13.16.2.14) to initiate the reprint of a group of labels.

To reprint:

- 1 Enter data in the header, specifying by range.
- 2 Press Go.
The system displays the labels that satisfy the criteria.
- 3 In the displayed data, change No to Yes in the Sel column to print the label.
- 4 Press Go.

Fig. 3.11
Manual Label Reprint (36.13.16.2.14)

The screenshot shows the 'Manual Label Reprint' application window. The top part is a form with various input fields for filtering labels. Below the form, there is a table titled 'Labels to Release' with columns for 'Sel', 'Format', 'Label ID', 'Date', 'Request ID', and 'M'. The 'Sel' column contains 'yes' for the first row and 'no' for the others. An arrow points from the 'Output: Batch ID:' field in the form to the table.

Sel	Format	Label ID	Date	Request ID	M
yes	CITEM	6	2/25/2015	7	
no	CITEM	56	2/25/2015	74	
no	CITEM	57	2/25/2015	76	
no	CITEM	7	2/25/2015	9	
no	CITEM	108	4/29/2015	144	
no	CITEM	129	1/5/2016		
no	CITEM	136	1/5/2016		
no	CITEM	137	1/5/2016		
no	CITEM	138	1/5/2016		
no	CONTNR	86	2/25/2015	113	
no	CONTNR	9	2/25/2015	12	
no	CONTNR	14	2/25/2015	18	
no	CONTNR	2	2/25/2015	2	
no	CONTNR	19	2/25/2015	24	
no	CONTNR	26	2/25/2015	39	

Pre-Select All. Specify Yes to have the system set the Sel column to Yes, thereby selecting to reprint all labels.

From/To Date. Specify a range of print request dates.

From/To Format ID. Specify a range of valid format IDs as defined in Label Format Maintenance (36.13.16.1.9).

From/To Request ID. Specify a range of print request IDs.

From/To App Request ID. Specify a range of source application request IDs.

From/To Label. Specify a range of labels.

From/To App. Specify a range of applications. This is the name of the source application sent from the calling programs as a parameter.

From/To Domain. Specify a range of domains from which to print.

From/To Purpose. Specify a range of purposes to print. This is user-defined by the calling application. It can be blank or populated with the data from the caller, depending on whether you send it using an API.

From/To Label Key 1- Label Key 3. Specify a range of key labels.

Ad Hoc Label Generation

Use Ad Hoc Label Generation (36.13.16.2.19) to create labels based on the definitions set up in Label Format Maintenance (36.13.16.1.9).

Note You can access functions similar to Ad Hoc Label Generation through the Label Print Status Collection by right-clicking on a label in the top browse and selecting Label Reprint/Ad Hoc Label Generation. When you do, you can reprint or create new labels based on existing ones. When you keep original values, the system reprints the label. When you modify values, the system creates a brand new label; see Label Print Status Collection.

Copy Label Data from Existing Labels to Save Time

When you select a label format in this program, the system prompts you to copy data from an existing label ID. When you choose Yes, the system displays a frame that lists all label master records that were previously generated using the label format you specified. You can optionally apply a filter to find the label to copy, using the fields Label Keys 1, 2, and 3. (For information on Label Keys, refer to the Key field described in “Label Maintenance” on page 58.)

Once you select the label to copy, the system prompts you to enter the printer and number of copies. When the system copies data to a new label, the system populates the fields in the Value column, using the data from the label ID you selected. You can optionally modify the data. You can enter a value since the value is blank; when you do not enter a value, the label prints with a blank value.

When you retain the original label data, the system executes a reprint of the label. When you modify data, the system prints the new label.

When you choose not to copy a label, the system continues through normal processing.

Navigation

Select a label template for which you want to enter data to produce a label. Enter it in the Label Format field; then select a printer.

After you specify the template and printer, you can specify the number of labels to print and enter additional data--such as such as item numbers, color, location, and so on-- into the template fields that display for the specified label format.

You can enter up to 500 characters in the Variable Value field. When you enter the field to enter new values or edit existing value, the system displays a separate frame for better visibility for longer values.

Specify the number of copies of the label to print in the Copies field.

Press Go to print the number of labels you specified in the Copies field, using the data you entered.

Fig. 3.12
Ad Hoc Label Generation (36.13.16.2.19)

Field Name	Value
tt-CpItem.cp_cust_part	
tt-CpItem.CtryOfOrigin	
tt-CpItem.labelFormatID	
tt-CpItem.LabelQty	
tt-CpItem.LabelUserId	
tt-CpItem.pt_desc1	
tt-CpItem.pt_desc2	

Label Format. Indicate the label format as defined in Label Format Maintenance (36.13.16.1.9).

Printer. Indicate a valid system printer. The system validates for defined syntax for a printer. The system prompts you for re-entry when you enter an invalid value.

Syntax Type. Indicate a valid syntax type as defined in Label Printer Syntax Maint (36.13.16.1.2).

Field Name. Select a field name. The system opens the field value where you can place the value you are replacing in the label name.

Value. Indicate a valid value for the field. You can enter up to 500 characters in a separate frame that displays.

Automatically Start and Release Requests from Queues

Use Label Auto Release Startup (36.13.16.2.16) to have the system automatically start and release label print requests from queues. You indicate the time in seconds that you want the system to ignore the requests. After the specified time, the system automatically releases the requests for processing.

Fig. 3.13
Label Auto Release Startup (36.13.16.2.16)

Label Auto Release Startup x

Sleep seconds: 0 Redraw delay: 10

Pertinent fields for label printing include:

Sleep seconds. Indicate the time in seconds the system waits before automatically processing requests.

Redraw delay. Indicate the time in seconds for the screen to wait to do a refresh. The system uses this value to determine when the screen redraws, preventing you from exiting, then re-entering to view data. In short, it causes the auto-refresh of the screen.

Once you enter the sleep and redraw delay time and press Enter, the system displays a frame for you to enter a queue list name and displays the status. See Figure 3.14.

Fig. 3.14
Auto Release Startup, Queue List and Status

Label Auto Release Startup x

Queue List	Status	Last Run Dat
█		

Queue List. Enter a valid queue name. The system prompts you to indicate whether a print server on the server for the queue should be started.

Status. Displays the current status of the AppServer agent. Possible values are:

- SLEEPING
- PROCESSING

To stop the agent, type `END` in the Status field.

Last Run Date. The system displays the last time auto release ran.

Making Calls to the Label Printing Services

Calls to QAD Label Printing Services are required when you want to use the service. You must make code changes to perform the call. All calls are made to Label Printing Services by setting the c-application-mode variable to `API` in the calling program. Set the variable to `API` before calling Label Printing Services.

The following lists several ways that you can call the QAD Label Printing Services:

A simple API signature with the caller determining the format and printer is as follows:

```
define input parameter icReqID as character no-undo. /* REQ ID*/
define input parameter icPrintType as character no-undo. /* PRINT or RE */
```

```

define input parameter icPrinterId as character no-undo. /* PRINTER ID */
define input parameter icFormatId as character no-undo. /* FORMAT */
define input parameter iNbrOfCopy as integer no-undo. /* # of Copies */
define input parameter dataset-handle ipPrintDataset. /*DataSet Handle*/
define output parameter table for temp_err_msg. /* ERR MSGS */

```

Where:

Dataset can be any dataset; however, the system generates a label for each looping outer dataset.

Use the following information to help you make the call, which supports both Progress Dataset and XML as data source:

- Using Standard Configuration (Dataset as data source):

```

RUN us/lb/lbprteng.p(
  INPUT request_id, /* REQ_ID */
  INPUT print_type, /* Print Type - Print or Reprint */
  INPUT printer, /* Printer */
  INPUT lbl_format, /* Label Format to use */
  INPUT 2, /* Number of Copies */
  INPUT dataset-handle hDataSet, /* Dataset handle containing data */
  output table temp_err_msg). /* Result in the form of temp-table */

```

- Using Application Configuration (XML as data source):

```

RUN us/lb/lbpeapxm.p(
  INPUT request_id, /* Request ID */
  INPUT "EAM", /* APPLICATION ID */
  INPUT "10C1000", /* Context1 */
  INPUT "", /* Context2 */
  INPUT xmlStream, /* XML stream containing data */
  output table temp_err_msg). /* Result in the form of temp-table */

```

- Using application configuration:

```

RUN us/lb/lbpeapds.p(
  INPUT request_id, /* Request ID */
  INPUT "EAM", /* APPLICATION ID */
  INPUT "10C1000", /* Context1 */
  INPUT "", /* Context2 */
  INPUT dataset-handle hDataset, /* Dataset handle containing data */
  output table temp_err_msg). /* Result in the form of temp-table */

```

- Using content-based routing pre-processor:

```

RUN us/lb/lbcrpp.p(
  INPUT "Build-Shipper", /* PURPOSE */
  INPUT DATASET-HANDLE phDataset1 BY-REFERENCE, /* DATASET CONTAINING DATA */
  OUTPUT TABLE temp_err_msg). /* Result in the form of temp-table */

```

Using Labels as Templates

This chapter discusses the following topics:

***Label Overview* 68**

Introduces importing and using labels as templates

***Import Label Data* 69**

Tells you how to import the out-of-the-box Label Printing Services labels.

***Configuration* 74**

Provides information on configuring the out-of-the-box labels.

***Extraction Programs* 75**

Provides an overview and discusses pertinent aspects of extraction programs.

***SO Labels* 83**

Provides a table of label configuration and an illustration of each label, with markers of data. Also, provides a cross-reference table that lists LPS fields and other data for each marked label.

***DRP Labels* 90**

Provides the same information as for the SO Labels section; however, the information pertains to distributed order (DO)-related labels for use in QAD distributed requirements planning (DRP) processes.

Label Overview

QAD provides a set of pre-designed labels that you can use as templates for your own labels.

You first import the labels in your system; see “Import Label Data” on page 69. Once imported, you use the labels as templates. Information in this chapter includes graphics of each label and a breakdown of the data on the label that corresponds to the Label Printing Service program, event, field, and so on that configures the label data.

Important When changes to the labels are required, please contact QAD Services for help in personalizing the labels for your business requirements. The labels provided are templates, and they are not supported by QAD.

You can print the labels using:

- QAD Automation Solutions: Data Collections transactions
- Custom programs
- Standard QAD Serialization menus that have label print functions; however, you can only print labels with the serial label format

Refer to documentation for QAD Automation Solutions: Data Collections or QAD Serialization for more information.

Label Specifications

All labels to be used as templates have the following specifications:

- 203 dpi
- 4x6 inch dimensions
- Work only with Zebra printers only (Model GKT420T-200 and QLN420)
- Available in Label Printing Services, version 3.1.2 or higher

Label Descriptions

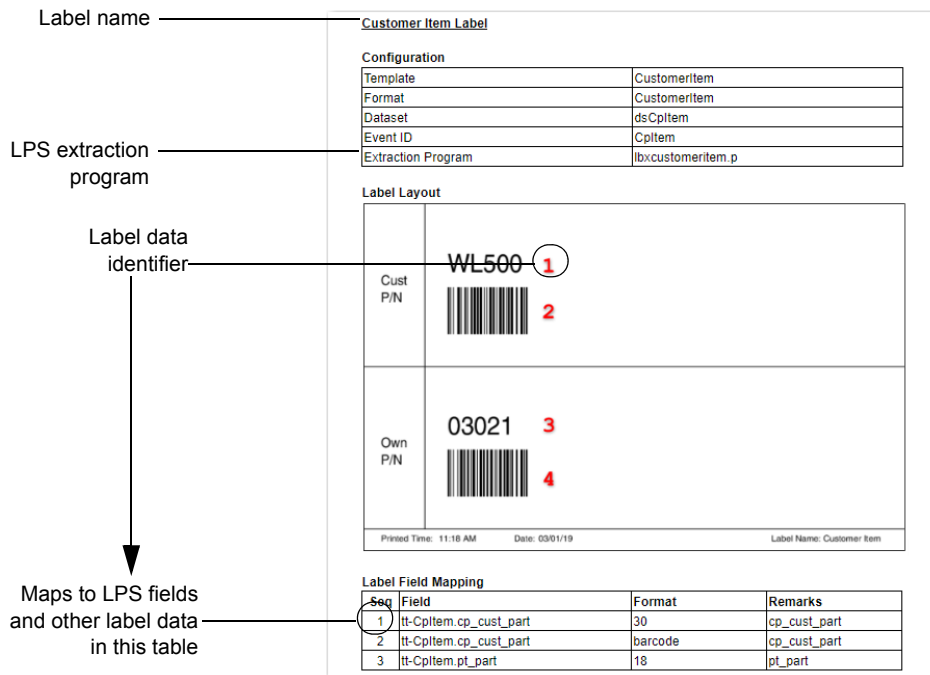
The label information in this chapter is set up by label name, such as Customer Item Label. For each label provided as a template, the subsections of the labels include:

- Label configuration
A table of the label, depicting the template name, format, dataset, event ID, and the program that extracts the label data
- Label layout
An illustration of the label with numbered callouts of data on the label
- Label Field Mapping
A table with numbers that correspond to the callouts in the illustration. The table includes the field in the extract program that holds the data, the format of the data, and any remarks.

Note For some mappings, parts of the label contain multiple pieces of mapped data within a single marked area. For example, if a barcode area is marked as number 7, the field mapping may include a 7.1, 7.2, and so on in the field mapping.



Fig. 4.1
Label Data Layout



Import Label Data

Before you use the labels as templates for your needs, you:

- Import Labels
- Import Printer
- Import Format
- Import Extraction Configuration

All label configuration is in the following directory:

```
${DIST}/as-setup/labels
```

Import Labels

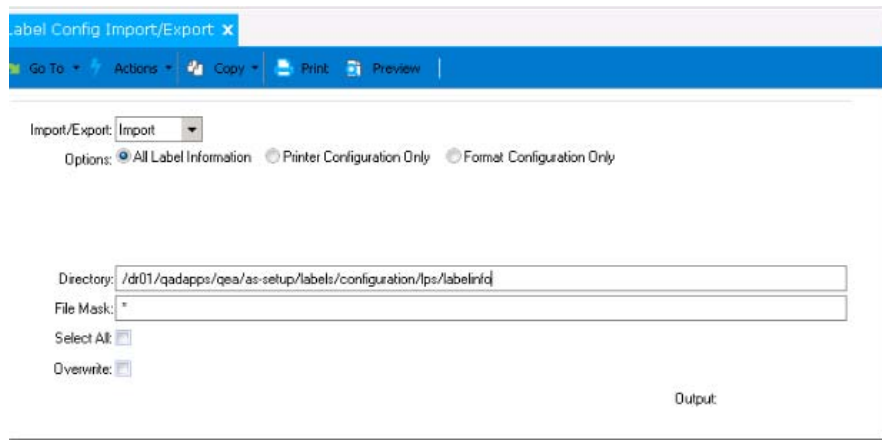
For a new implementation, follow these steps to import the labels:

- 1 Create a `template` directory to hold label templates; then, move the contents from `${DIST}/as-setup/labels/templates/lps` to the new template directory.
- 2 Create a `dataset` directory to hold label datasets; then, move the contents from `${DIST}/as-setup/labels/datasets/lps` to the new dataset directory.
- 3 Go on to step 4 on page 70 for an existing environment.

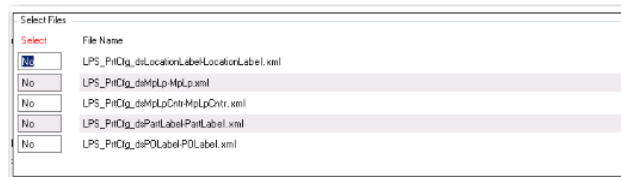
For an existing implementation, follow these steps to import the labels:

- 1 Locate the template and dataset directories for the environment.
- 2 Move the contents from $\${DIST}/as-setup/labels/templates/lps$ to the template directory located in Step 1.
- 3 Move the contents from $\${DIST}/as-setup/labels/datasets/lps$ to the dataset directory located in Step 1.
- 4 Use Label Config Import/Export (36.13.16.4.1) to import label data. Set fields and options in the program as shown in the following table; then, press Return.

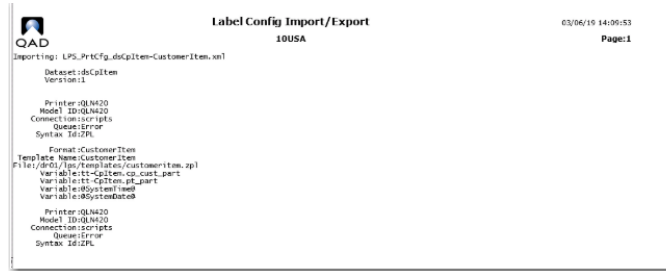
Field	Set To:
Import/Export	Import
Options	All label information
Directory	$\${DIST}/as-setup/labels/configuration/lps/labelinfo$
File Mask	* (or specific part or complete file name to import)
Select All	Click to set to Yes
Overwrite	For an existing environment, click to set to Yes when you want to overwrite existing data with the contents from the import file.



- 5 When the system displays the frame of files, select the files to import; then, press Go:



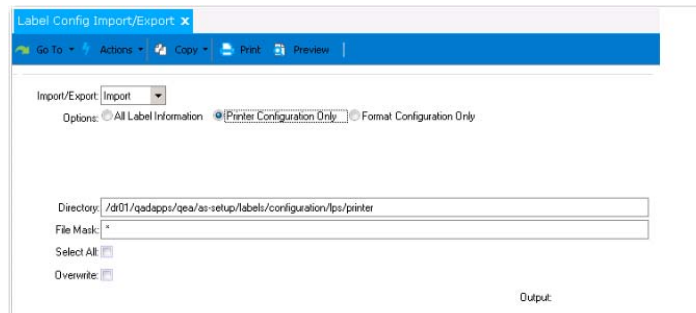
When the import completes, the system reports imported files:



Import Printer

To import the printer, you use the same program as you did to import labels:

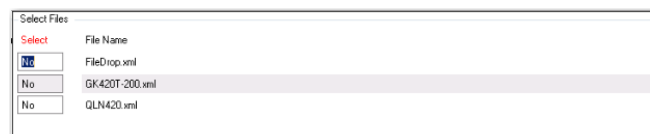
- 1 Run Label Config Import/Export.
- 2 For Options, select Printer Configuration Only:



- 3 Set other options in the program as shown in the following table; then, press Return.

Field	Set To:
Import/Export	Import
Options	Printer Configuration Only
Directory	$\${DIST} / as - setup / labels / configuration / lps / printer$
File Mask	* (or specific part or complete file name to import)
Select All	Check to select all files to import
Overwrite	Check to overwrite the contents from import file over the existing data in the environment

- 4 Select the files to import; then press Go:



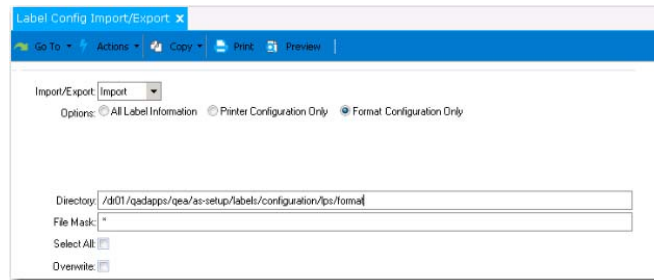
When the import completes, the system reports imported files:



Import Format

To import the format, you use the same program as you did to import labels:

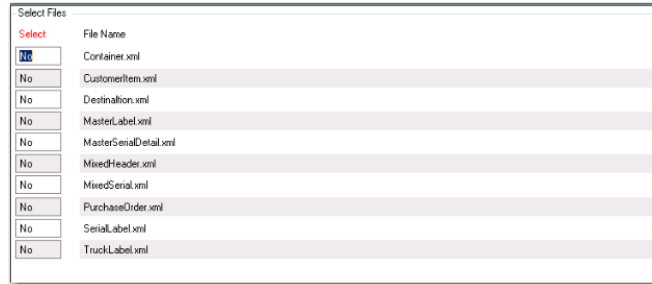
- 1 Run Label Config Import/Export.
- 2 For Options, select Format Configuration Only:



- 3 Set other options in the program as shown in the following table; then, press Return.

Field	Set To:
Import/Export	Import
Options	Format Configuration Only
Directory	`\${DIST}/as-setup/labels/configuration/lps/format
File Mask	* (or specific part or complete file name to import)
Select All	Check to select all files to import
Overwrite	Check to overwrite the contents from import file over the existing data in the environment

- 4 Select the files to import; then press Go:



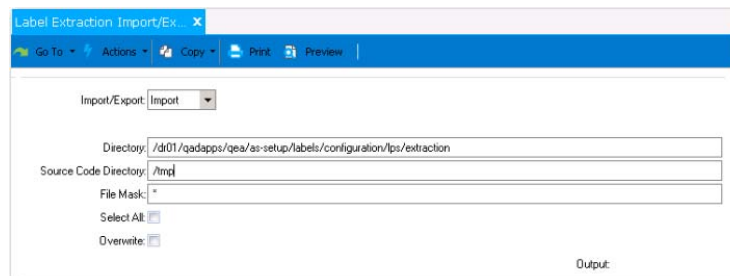
When the import completes, the system reports imported files:



Import Extraction Configuration

To extract configuration data, you use Label Extraction Import/Export (36.13.16.4.2).

- 1 Run Label Extraction Import/Export.
- 2 For Options, select Printer Configuration Only:

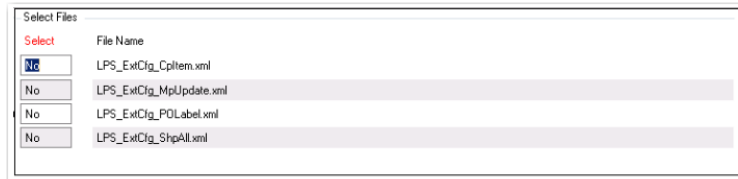


- 3 Set options in the program as shown in the following table; then, press Return.

Field	Set To:
Import/Export	Import
Options	Printer Configuration Only
Directory	<code>\${DIST}/as-setup/labels/configuration/lps/extraction</code>

Source Code Directory	tmp directory Note: The system dumps extraction source code to the tmp directory; however, the system ignores the directory.
File Mask	* (or specific part or complete name of file name to import)
Select All	Check to select all files to import
Overwrite	Check to overwrite the contents from import file over the existing data in the environment

4 Select the files to import; then, press Go:



When the import completes, the system reports imported files:



Configuration

You can route all out-of-the box labels for the following fields of label data:

- Printer
- Device
- Site
- Warehouse
- Area

The values are provided to the labels through the label event workfile. The following label event workfile fields are mapped in the extraction program to a specific value in label data:

- Key1 Event ID
- Key2 Warehouse code
- Key3 Area code
- Key4 Printer name

- Key5 Source Application name
- Key6 Device ID

Note For information on importing the extraction program, see “Import Extraction Configuration” on page 73.

Templates for content and printer routing setup for each label are available in the LPS package. You import the configuration provided in the package to use the out-of-the-box setup, using Label Config Import/Export (36.13.16.4.1) to select the label, printer configuration, or label format setup data to export from or import to a specific environment.

You can choose the setup data to be imported by specifying all labels, printer configuration only, or format configuration only. You can also choose to select all and overwrite setup data in the environment.

All label configuration is in the following directory:

```
${DIST}/as-setup/labels
```

Extraction Programs

Overview

Label Printing Services uses programs to extract all necessary data relevant to a particular label request. These programs, known as extraction programs, then produce a dynamic dataset that gets passed in a call to the pre-parsing routing program, which looks at the content and printer routing tables to determine the format and printer to use in the label request. Lastly, the extraction program may call other extraction programs that are linked to child label types that belong to the same label group.

Extraction programs are specified when defining label types in Label Type Maintenance (36.13.16.3.1), and the key table(s) and fields used to drive the data extraction process are specified when defining label events in Label Event Maintenance. Both maintenance programs are a part of the Label Extraction Configuration menu inside Label Printing Services. Below are the minimum include files and input parameters required to execute the extraction programs. Additionally, the tables in this section contain general information regarding all of the extraction programs included in the Label Printing Services 3.3 package.

Include Files

```
us/bbi/mfdeclre.i
us/bbi/gplabel.i
us/px/pxpgmmgr.i
us/px/pxphdef.i lblstdxr
us/px/pxgetph.i lblstdxr
us/px/pxphdef.i lbevntwkflxr
us/px/pxgetph.i lbevntwkflxr
us/lb/lbtcanextract.i
us/lb/lbds/lbevntwkfl.i
```

Input Parameters

The following lists the input parameters associated with the extraction programs. The table depicts the parameter, the type (decimal, and so on), and a brief description. (For more information on input parameters for the Extraction program, refer to the QAD Automation Solutions: Label Printing Services training material in the QAD Document Library.)

Table 4.1
Extraction Program Input Parameters

Input	Type	Description
Label event workfile ID	Decimal	The unique record ID of the label event workfile record generated during the incoming label request
Label request ID	Int64	The request ID of the incoming label request
Label group record ID	Decimal	These are the unique record IDs for the associated label group, label event processing, label event, and label group type records referenced during the processing of the label request
Label event processing record ID	Decimal	
Label event record ID	Decimal	
Label group type ID	Decimal	
Query handle object	Handle	The handle object for the query which is used to drive the data extraction process for the label
Current query row	Integer	The current row or iteration of the incoming query
Field values matched to field names in Label Event Maintenance	Character	This is for the field values passed in from the label request that correspond to the field names specified in Label Event Maintenance for the event referenced; these values will be combined into a single character value.
Catch-all value	Character	The value to be used as a catch-all so that when a label request comes in for a particular label type, that label is printed regardless of the label data. This value typically consists of a series of asterisks, one for each field name specified in Label Event Maintenance.
Extraction temp table	Table	This is the ttCanExtract temp-table to be used for both determining extractability and obtaining the label format and number of copies specified in Label Event Type Criteria Maintenance.

Provided Extraction Programs

The following table lists the extraction programs that are provided with the Label Printing Services package, version 3.3 or higher.

Table 4.2
Extraction Programs

Extraction Program	Purpose	Key Table Queried	Other Tables Queried	Dataset(s) Created
lboxshipper.p ^a	Extract shipment data and request all child label prints for shipper label	abs_mstr	serd_det, ser_mstr ser_mstr	None
lboxpassthrough.p ^b	Group labels together before sending for print	None	None	None



lbxmstrlpcontainer.p	Extract serial container detail and request print for container label	serd_det	serd_det, ser_mstr, abs_mstr, so_mstr, sod_det, pt_mstr, cp_mstr, ad_mstr, lbEvtWkfl	dsMpLpCntr
lbxmstrlpdetail.p	Extract serial detail and request print for serial mixed label	serd_det	serd_det, ser_mstr, abs_mstr, so_mstr, sod_det, pt_mstr, cp_mstr, ad_mstr, lbEvtWkfl	dsMpLpDetail
lbxmstrlp.p	Extract master serial data on shipper and request print for master license plate label	serd_det	serd_det, ser_mstr, abs_mstr, so_mstr, sod_det, pt_mstr, ad_mstr, cp_mstr, lbEvtWkfl	dsMpLp
lbxmstrlpdestination.p	Extract shipper destination address data and request print for destination label	serd_det	serd_det, ser_mstr, so_mstr, sod_det, abs_mstr, pt_mstr, ad_mstr, cp_mstr, edtar_mstr, edtmx_ref, lbEvtWkfl	dsMpLpDest
lbxmstrlpmix.p	Extract serial detail and request print for serial mixed label	serd_det	serd_det, ser_mstr, abs_mstr, so_mstr, sod_det, pt_mstr, ad_mstr, cp_mstr, lbEvtWkfl, edtar_mstr, edtmx_ref	dsMpLpMix
lbxmstrlpmstr.p	Extract serial detail and request print for serial master label	serd_det	ser_mstr, serd_det, abs_mstr, so_mstr, sod_det, pt_mstr, ad_mstr, cp_mstr, lbEvtWkfl, edtar_mstr, edtmx_ref	dsMpLpMstr
lbxdomstrlpmix.p	Extract serial detail and request print for distribution order serial mixedLabel	See lbxdomstrlpmixmstr.i		
lbxdomstrlpmstr.p	Extract serial detail and request print for distribution order serial master label	See lbxdomstrlpmixmstr.i		
lbxdomstrlpmixmstr.i	Handles the label data processing for both lbxdomstrlpmix.p and lbxdomstrlpmstr.p	serd_det	ser_mstr, serd_det, dsd_det, ds_det, abs_mstr, dss_mstr, pt_mstr, ad_mstr, cp_mstr, lbEvtWkfl, edtar_mstr, edtmx_ref	dsDOMpLpMix, dsDOMpLpMstr
lbxdomstrlpdetail.p	Extract serial detail and request print for distribution order serial detail label	serd_det	serd_det, ser_mstr, abs_mstr, dsd_det, ds_det, dss_mstr, pt_mstr, cp_mstr, ad_mstr, lbEvtWkfl	dsDOMpLpDetail
lbxdomstrlpmstrdest.p	Extract shipper destination address data and request print for distribution order destination label	serd_det	serd_det, ser_mstr, abs_mstr, dsd_det, ds_det, dss_mstr, pt_mstr, ad_mstr, cp_mstr, lbEvtWkfl, edtar_mstr, edtmx_ref	dsDOMpLpMstrDest

lboxdomstrlpcontainer.p	Extract serial container detail and request print for distribution order container label	serd_det	serd_det, ser_mstr, abs_mstr, dsd_det, ds_det, dss_mstr, pt_mstr, lbEvtWkfl, ad_mstr, edtar_mstr, edtmx_ref, cp_mstr	dsDOMpLpCntr
lboxcustomeritem.p	Extract customer item data and request print for customer item label	pt_mstr	ad_mstr, pt_mstr, so_mstr, sod_det, cp_mstr, lbEvtWkfl, cp_mstr, edtar_mstr, edtmx_ref	dsCplItem
lboxpurchaseorder.p	Extract purchasing data and request print for purchase order label	pod_det	pod_det, po_mstr, lbevntwkfl, pt_mstr, ad_mstr, vp_mstr	dsPOLabel
lboxserialprint.p	Extract serial data and request print for serial Label	ser_mstr	ser_mstr, lbEvtWkfl, pt_mstr, si_mstr, loc_mstr, LocM, WhseM, ad_mstr, ld_det, bom_mstr, pckc_mstr, serd_det, wo_mstr, abs_mstr, pod_det, po_mstr	dsSerialPrintData
lboxseqblrk.p	Extract serial data and request print for serial Label	abs_mstr	abs_mstr, ad_mstr, serd_det, ser_mstr, abss_det, pt_mstr, lbEvtWkfl	dsBlend, dsRack

a. lboxshipper.p is used to query for general shipper (abs_mstr) and serial (ser_mstr, serd_det) data, which it passes to its child group types to handle the more detailed extraction logic; therefore, no datasets are created in this program.

b. lboxpassthrough.p is used strictly for grouping purposes. The program simply passes the information from its input parameters to its child group types to handle the more detailed extraction logic; therefore, tables are not queried, and datasets are not created in this program.

Labels

The section includes the LPS out-of-the-box generic labels that you import and use as templates. Each subsection names the label, provides a graphic of the label, and includes tables of label information. This section contains the following labels:

- Customer Item Label
- Purchase Order Label
- Serial Label
- Rack Label
- Blend Label

Note Refer to SO Labels for sales-order-related labels and to DRP Labels for distributed-order-related labels.

Customer Item Label

Table 4.3 Customer Item Label Configuration

Template	CustomerItem
Format	CustomerItem
Dataset	dsCpItem
Event ID	CpItem
Extraction Program	lboxcustomeritem.p

Fig. 4.2
Customer Item Label



Table 4.4
Customer Item Label Field Mapping

Seq	Field	Format	Remarks
1	tt-CpItem.cp_cust_part	30	cp_cust_part
2	tt-CpItem.cp_cust_part	barcode	cp_cust_part
3	tt-CpItem.pt_part	18	pt_part
4	tt-CpItem.pt_part	barcode	pt_part

Purchase Order Label

Table 4.5
Purchase Order Label Configuration

Template	PurchaseOrder
Format	PurchaseOrder
Dataset	dsPOLabel
Event ID	POLabel
Extraction Program	lboxpurchaseorder.p

Fig. 4.3
Purchase Order Label

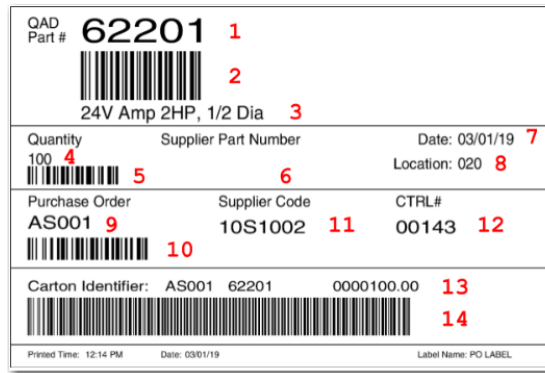


Table 4.6
Purchase Order Label Field Mapping

Seq	Field	Format	Remarks
1	tt-POLabel.pod_part	18	pod_part
2	tt-POLabel.pod_part	barepde	pod_part
3	tt-POLabel.pt_partdesc	49	Pt_desc1 + “ “ + pt_desc2
4	tt-POLabel.LabelQty	->>>9.99	lbEvtWkfl.decflld
5	tt-POLabel.LabelQty	barcode	lbEvtWkfl.decflld
6	tt-POLabel.pod_vpart	30	pod_vpart
7	SystemDate	99/99/99	System Date
8	tt-POLabel.pod_loc	8	pod_loc
9	tt-POLabel.pod_nbr	8	pod_nbr
10	tt-POLabel.pod_nbr	barcode	pod_nbr
11	tt-POLabel.po_vend	8	po_vend
12	tt-POLabel.ControlID	5	Last five digits of lblEngRequestID
13	tt-POLabel.CartonIdent	36	Pod_nbr + pod_part + lbEvtWkfl.decflld
14	tt-POLabel.CartonIdent	36	Pod_nbr + pod_part + lbEvtWkfl.decflld

Serial Label

Table 4.7
Serial Label Configuration

Template	SerialLabel
Format	SerialLabel
Dataset	dsSerialPrint
Event ID	SerialPrint
Extraction Program	lbxserialprint

Fig. 4.4
Serial Label

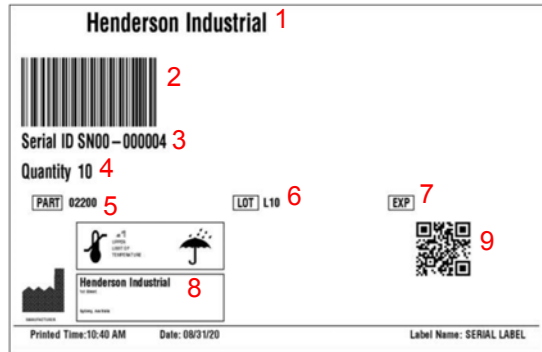


Table 4.8
Serial Label Field Mapping

Seq	Field	Format	Remarks
1	tt_ser_label.tt_Supplier_ad_name	30	ad_name
2	tt_ser_label.tt_ser_serial_id	barcode	ser_serial_id
3	tt_ser_label.tt_ser_serial_id	40	ser_serial_id
4	tt_ser_label.tt_ser_qty_avail	->,>>>,>>9.9<<<<<<<<<	Serial quantity
5	tt_ser_label.tt_pt_part	18	Serial part
6	tt_ser_label.tt_ld_lotser	18	ser_lotser
7	tt_ser_label.tt_ld_expire	99/99/99	ld_expire
8	tt_ser_label.tt_Supplier_ad_name	99/99/99	System Date
8.1	tt_ser_label.tt_Supplier_ad_line1	36	ad_line1
8.2	tt_ser_label.tt_Supplier_ad_line2	36	ad_line2
8.3	tt_ser_label.tt_Supplier_ad_line3	36	ad_line3
8.4	tt_ser_label.tt_Supplier_ad_city	20	ad_city
8.5	tt_ser_label.tt_Supplier_ad_state	4	ad_state
8.6	tt_ser_label.tt_Supplier_ad_zip	10	ad_zip
9	01tt_ser_label.tt_pt_part10tt_ser_label.tt_ld_lotser21tt_ser_label.tt_ser_serial_id	QR code	01 + ser_part + 10 + ser_lotser + 21 ser_serial_id

Rack Label

Table 4.9
Rack Label Configuration

Template	Rack
Format	Rack
Dataset	dsRack
Event ID	Rack
Extraction program	lhxseqblk.p

Fig. 4.5
Rack Label

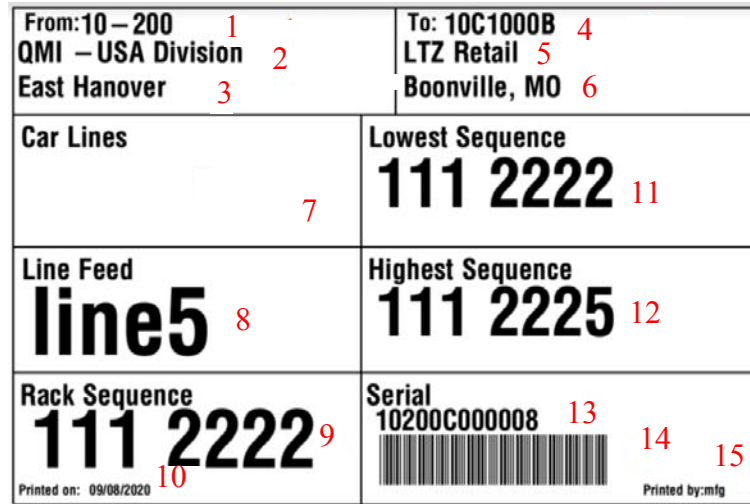


Table 4.10
Rack Label Field Mapping

Seq	Field	Format	Remarks
1	ttRack.fromAddressLine1	x(8)	ad_mstr.ad_addr
2	ttRack.fromAddressLine2	x(36)	ad_mstr.ad_name
3	ttRack.fromAddressLine3	x(20)	ad_mstr.ad_city
4	ttRack.toAddressLine1	x(8)	ad_mstr.ad_addr
5	ttRack.toAddressLine2	x(36)	ad_mstr.ad_name
6	ttRack.toAddressLine3	x(26)	ad_mstr.ad_city + ad_mstr.ad_state
7	ttRack.carLines	x(30)	abss_det.abss_dock
8	ttRack.lineFeed	x(30)	abss_det.abss_line_feed
9	ttRack.lowestSequence	x(30)	abss_det.abss_cust_seq
10	ttRack.date	99/99/9999	Today
11	ttRack.lowestSequence	x(30)	abss_det.abss_cust_seq
12	ttRack.highestSequence	x(30)	abss_det.abss_cust_seq
13	ttRack.serial	x(40)	ser_mstr.ser_serial_id
14	ttRack.serial	barcode	ser_mstr.ser_serial_id
15	ttRack.userID	x(8)	lbEvtWkfl.modUserid

Blend Label

Table 4.11
Blend Label Configuration

Template	Blend
Format	Blend
Dataset	Blend
Event ID	Blend
Extraction program	lhxseqblk.p

Fig. 4.6
Blend Label

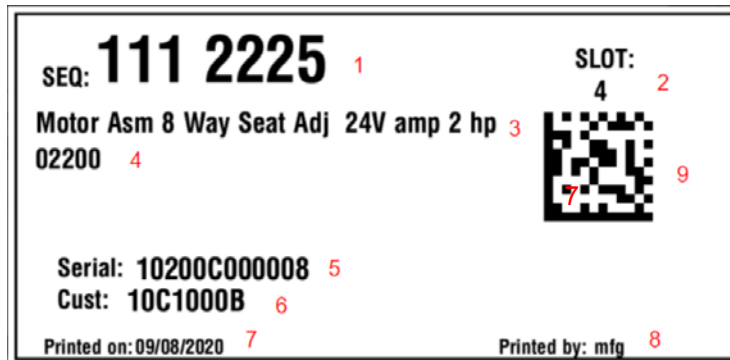


Table 4.12
Blend Label Field Mapping

Seq	Field	Format	Remarks
1	ttBlend.sequence	x(30)	abss_det.abss_cust_seq
2	ttBlend.slot	x(8)	Sequence number of Customer Sequence
3	ttBlend.itemDescription	x(50)	pt_mstr.pt_desc1 + pt_mstr.pt_desc2
4	ttBlend.itemNumber	x(18)	pt_mstr.pt_part
5	ttBlend.serial	x(40)	ser_mstr.ser_serial_id
6	ttBlend.abs_shipto	x(8)	abs_mstr.abs_shipto
7	ttBlend.date	99/99/9999	Today
8	ttBlend.userID	x(8)	lbEvntWkfl.modUserid
9	ttBlend.sequence	Barcode	abss_det.abss_cust_seq

SO Labels

The following labels are related to sales orders (SO):

- Master Label
- Mixed Label
- Container Label
- Destination Label
- Serial Detail Label
- Truck Label

Master Label

Table 4.13
Master Label Configuration

Template	MasterSerial
Format	MasterLabel
Dataset	dsMpLpMstr

Mixed Label

Table 4.15
Mixed Label Configuration

Template	MixedLabel
Format	MixedSerial
Dataset	dsMpLpMix
Event ID	MpUpdate, ShpAll (part of group of labels)
Extraction Program	lbxmstrlpmix.p

Fig. 4.8
Mixed Label

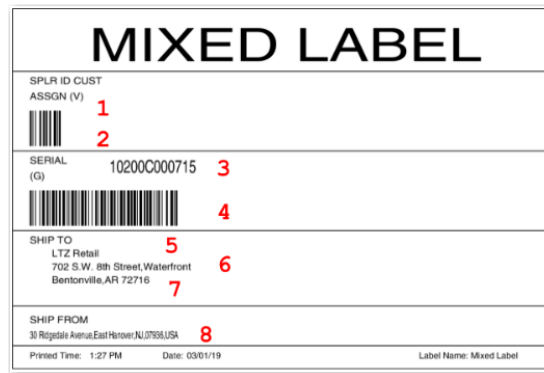


Table 4.16
Mixed Label Field Mapping

Seq	Field	Format	Remarks
1	tt-MpLpMix.EDTMXSuppNo	8	edtmx_ref.edtmx_tp_addr Search based on edi inbound document for edtmx_tp_doc = "856" or "DESADV"
2	tt-MpLpMix.EDTMXSuppNo	barcode	edtmx_ref.edtmx_tp_addr Search based on edi inbound document for edtmx_tp_doc = "856" or "DESADV"
3	tt-MpLpMix.ser_serial_id	40	ser_serial_id
4	tt-MpLpMix.ser_serial_id	barcode	ser_serial_id
5	tt-ShipTo.ad_name	36	ad_name
6	tt-ShipTo.ad_oneplustwo	73	ad_line1 + "," + ad_line2
7	tt-ShipTo.ad_citystzip	34	ad_city + ad_state + ad_zip
8	tt-ShipFrom.ad_completeaddr	114	ad_line1 + ad_line2 + "," + ad_city + "," + ad_state + "," + ad_zip + "," + ad_etry

Container Label

Table 4.17 Container Label Configuration

Template	Container
Format	Container
Dataset	dsMpLpCntr
Event ID	MpUpdate, ShpAll (part of group of labels)
Extraction Program	lbxmstrlpcontainer.p

Destination Label

Table 4.19
Destination Label Configuration

Template	Destination
Format	Destination
Dataset	dsMpLpDest
Event ID	MpUpdate, ShpAll (part of group of labels)
Extraction Program	lbxmstrlpdestination.p

Fig. 4.10
Destination Label


PLANT LOCATION CODE 1	DOCK LOCATION 2
PLANT LOCATION CODE (1L)  4	
SHIP TO LTZ Retail 702 S.W. 8th Street, Waterfront Bentonville, AR 72716 5	SHIP DATE 03/01/19 8
SHIP FROM 33 Ridgedale Avenue, East Hanover, NJ 07936, USA 9	
Printed Time: 1:27 PM Date: 03/01/19 Label Name: Destination	

Table 4.20
Destination Label Field Mapping

Seq	Field	Format	Remarks
1	tt-MpLpDest.EDTMXPlantCode	8	edtmx_ref.edtmx_tp_site Search based on edi inbound document for edtmx_tp_doc = "856" or "DESADV"
2	tt-MpLpDest.sod_dock	8	sod_dock
3	tt-MpLpDest.EDTMXPlantCode	8	edtmx_ref.edtmx_tp_site Search based on edi inbound document for edtmx_tp_doc = "856" or "DESADV"
4	tt-MpLpDest.EDTMXPlantCode	barcode	edtmx_ref.edtmx_tp_site Search based on edi inbound document for edtmx_tp_doc = "856" or "DESADV"
5	tt-ShipTo.ad_name	35	ad_name
6	tt-ShipTo.ad_oneplustwo	73	ad_line1 + "," + ad_line2
7	tt-ShipTo.ad_citystzip	34	ad_city + ad_state + ad_zip
8	SystemDate	99/99/99	System date
9	tt-ShipFrom.ad_completeaddr	114	ad_line1 + ad_line2 + "," + ad_city + "," + ad_state + "," + ad_zip + "," + ad_ctry

Table 4.24
Truck Label Field Mapping

Seq	Field	Format	Remarks
1	tt-MpLp.ser_serial_id	40	ser_serial_id
2	tt-MpLp.ser_serial_id	barcode	ser_serial_id
3	tt-MpLp.ser_ship_wt	>>, >>9.99<<<<	ser_ship_wt
4	tt-ShipTo.ad_name	36	ad_name
5	tt-ShipTo.ad_oneplustwo	73	ad_line1 + “,” + ad_line2
6	tt-ShipTo.ad_citystzip	34	ad_city + ad_state + ad_zip

DRP Labels

The section includes the LPS out-of-the-box labels that you import and use as templates. Each subsection names the label, provides a graphic of the label, and includes tables of label information. This section contains the following labels:

- DO Container Label
- DO Destination Label
- DO Master Label
- DO Master Serial Detail
- DO Mixed Serial Label
- DO Truck Label

DO Container Label

Table 4.25
DO Container Label

Template	DOContainer
Format	DOContainer
Dataset	dsDOMpLpCntr
Event ID	MpUpdate, ShpAll (part of group of labels)
Extraction Program	lbxdomstrlpcontainer.p

Fig. 4.13
DO Container Label

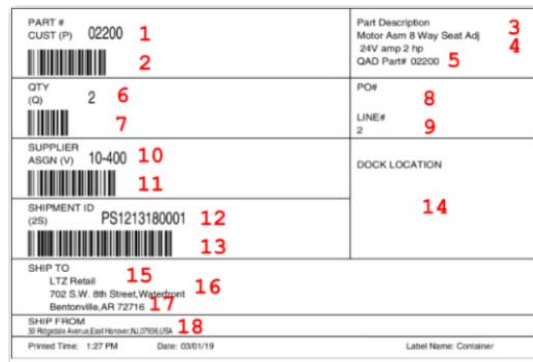


Fig. 4.14
DO Destination Label



Table 4.28
DO Destination Label Field Mapping

Seq	Field	Format	Remarks
1	tt-DOMpLpMstrDest.EDTMXPlant Code	8	edtmx_ref.edtmx_tp_siteSearch based on edi inbound document for edtmx_tp_doc = "856" or "DESADV"
2	tt-DOMpLpMstrDest.TADDockCode	8	edtar_tad_itm_val
3	tt-DOMpLpMstrDest.EDTMXPlant Code	8	edtmx_ref.edtmx_tp_siteSearch based on edi inbound document for edtmx_tp_doc = "856" or "DESADV"
4	tt-DOMpLpMstrDest.EDTMXPlant Code	barcode	edtmx_ref.edtmx_tp_siteSearch based on edi inbound document for edtmx_tp_doc = "856" or "DESADV"
5	tt-ShipTo.ad_name	36	ad_name
6	tt-ShipTo.ad_oneplustwo	73	ad_line1 + "," + ad_line2
7	tt-ShipTo.ad_citystzip	34	ad_city + ad_state + ad_zip
8	SystemDate	99/99/99	SystemDate
9	tt-ShipFrom.ad_completeaddr	114	ad_line1 + ad_line2 + "," + ad_city + "," + ad_state + "," + ad_zip + "," + ad_ctry

DO Master Label

Table 4.29
DO Master Label

Template	DOMasterSerial
Format	DOMasterLabel
Dataset	dsDOMpLpMstr
Event ID	MpUpdate, ShpAll (part of group of labels)
Extraction Program	lboxdomstrlpmstr.p

Table 4.34
DO Mixed Serial Label Field Mapping

Seq	Field	Format	Remarks
1	tt-DOMpLpMix.EDTMXSuppNo	8	edtmx_ref.edtmx_tp_addrSearch based on edi inbound document for edtmx_tp_doc = "856" or "DESADV"
2	tt-DOMpLpMix.EDTMXSuppNo	barcode	edtmx_ref.edtmx_tp_addrSearch based on edi inbound document for edtmx_tp_doc = "856" or "DESADV"
3	tt-DOMpLpMix.ser_serial_id	40	ser_serial_id
4	tt-DOMpLpMix.ser_serial_id	Barcode	ser_serial_id
5	tt-ShipTo.ad_name	36	ad_name
6	tt-ShipTo.ad_oneplustwo	73	ad_line1 + "," + ad_line2
7	tt-ShipTo.ad_citystzip	34	ad_city + ad_state + ad_zip
8	tt-ShipFrom.ad_completeaddr	114	ad_line1 + ad_line2 + "," + ad_city + "," + ad_state + "," + ad_zip + "," + ad_ctry

DO Truck Label

Table 4.35
DO Truck Label Configuration

Template	DOTruckLabel
Format	DOTruckLabel
Dataset	dsDOMpLpMstrDest
Event ID	MpUpdate, ShpAll (part of group of labels)
Extraction Program	lbxdomstrlpmstr.p

Fig. 4.18
DO Truck Label



Table 4.36
DO Truck Label Field Mapping

Seq	Field	Format	Remarks
1	tt-DOMpLpMstrDest.ser_serial_id	40	ser_serial_id
2	tt-DOMpLpMstrDest.ser_serial_id	Barcode	ser_serial_id
3	tt-DOMpLpMstrDest.ser_ship_wt	>>>>9.99<<<<	ser_ship_wt
4	tt-ShipTo.ad_name	36	ad_name

Seq	Field	Format	Remarks
5	tt-ShipTo.ad_oneplustwo	73	ad_line1 + “,” + ad_line2
6	tt-ShipTo.ad_citystzip	34	ad_city + ad_state + ad_zip



Index

Symbols

36.13.16.1.9 21
36.13.16.1.1 33

Numerics

36.13.16.1.1 7, 14, 36
36.13.16.1.2 7, 14
36.13.16.1.3 15
36.13.16.1.4 15
36.13.16.1.5 16
36.13.16.1.7 19
36.13.16.1.11 26
36.13.16.1.15 31
36.13.16.1.18 38
36.13.16.1.19 27
36.13.16.1.20 28
36.13.16.2.3 57
36.13.16.2.5 58
36.13.16.2.7 59
36.13.16.2.13 60
36.13.16.2.14 62
36.13.16.2.16 64
36.13.16.2.19 63
36.13.16.3.1. 40
36.13.16.3.4 41
36.13.16.3.13 42
36.13.16.3.14 43
36.13.16.3.15 43
36.13.15.4.1 45
36.13.16.4.1 46
36.13.16.4.2 47
36.13.16.1.12 26
36.13.16.1.13 29
36.19.1 13

A

Ad Hoc Label Generation 63
advanced contextual routing 7
API, making calls to print service 65
applications for label printing 29
AppServer connection 13
AppServer connection for label printing 13
AppServer Service Maintenance 13
auto delete data 22
auto release startup 64
auto-delete data 22

B

barcode label printing 3
barcode labeling functions 6

C

call without PROPATH 45
calling LPS 45, 65
calling program, setting variable 45, 65
calls to print service, API 65
c-application-mode variable in calling program 45, 65
configuration records, view 60
connect events with label groups 43
connection detail setup 7
content routing setup 33
contextual routing setup 30
copy function in Label Template Maintenance 19
create printer connections 16
CSS 6

D

Data Collection transactions 49
data tokens 8
data, extract 40
dataset, view and delete 26
datasets 26
datasets in XSD format 26
default printer types 15
delete datasets 26
delete label data 22
detailed architecture 4
details, connection 14
DO label templates 90
DRP label templates 90

E

event workfile 45, 65
events, setup 42
extract label data 40
extraction process 45
extraction programs
 input parameters 76
 list of programs 76
 overview 75

F

file drop connection 7
flat files 6
formats, for labels 18

G

grouping, labels 40
groups of labels 15

I

- import dataset for labels 26
- import templates 19
- import/export data 45
- import/export label data 46
- input parameters, extraction programs 76
- IPL 6

L

- label 13
 - configure applications 29
 - create groups 15
 - create printer connections 16
 - default printer types 15
 - define formats 18
 - import dataset 26
 - import templates 19
 - print overview 2
 - printer setup 14
 - set up connection details 7
 - setup for printing 12
 - syntax creation 7
 - syntax types 14
 - system tokens 8
 - third-party templates 6
 - track printing 8
- Label Application Configuration 29
- Label Auto Release Startup 64
- Label Config Import/Export 45, 46
- Label Content Routing Configuration 33
- Label Content Routing Setup 31
- label data, auto-delete 22
- label data, extract 40
- Label Dataset Import 26
- Label Dataset Maintenance 26
- Label Event Maintenance 42
- Label Event Processing 43
- Label Event Type Criteria Maintenance 43
- label event workfile 45, 65
- Label Event Workfile, Serialization 49
- Label Extraction Import/Export 47
- label format creation 21
- Label Format Maintenance 21
- Label Format Router Browse 60
- Label Group Maintenance 41
- label grouping 40
- Label Maintenance 58
- Label Manual Release 60
- Label Mapping 28
- Label Mapping Functions 27
- label mapping functions 27
- label print status 53
- Label Print Status Collection 53
- Label Printer Connection Maintenance 14
- Label Printer Connection Types 7
- Label Printer Maintenance 16
- Label Printer Router Browse 60
- Label Printer Routing Configuration 38
- label printer routing records 60
- Label Printer Routing Setup 36
- Label Printer Syntax Maintenance 14
- label printer, syntax types 14
- Label Printing Services 22
 - connection detail setup 14

- content routing setup 33
- contextual routing setup 30
- event setup 42
- extract data 40
- extraction process 45
- features 5
- import/export data 45, 46
- limitations 10
- make calls to 45
- map incoming data 28
- minimal configuration 12
- routing setup 36
- syntax examples 7
- tokens 8
- view status 53
- Label Printing Syntax 7
- Label Queue Monitor 57
- Label Reprint Maintenance 59
- label setup for Serialization 48
- Label Template Maintenance 19
- Label Type Maintenance 40
- labeling functions, barcode 6
- labels for Data Collections 49
- limitations for Label Printing Services 10

M

- Manual Label Reprint 62
- manually print labels 60
- map incoming data 28
- mapping, labels 27
- middleware 40
- minimal configuration 12

P

- PAS 6
- pint status, viewing 53
- print labels manually 60
- Print Queue Maintenance 15
- print queued labels 60
- print templates, third party 6
- printer hierarchy setup 36
- Printer Model Maintenance 15
- printers, setup for labels 14
- process flow 4
- program list, extraction programs 76

Q

- QAD Serialization
 - setting up labels
 - Serialization, label setup 48
 - queued labels, view 57
- QXtend 26

R

- release, auto startup 64
- reprint group labels 62
- reprint labels 59
- reprint labels from existing setup 63
- reprint labels, from browse collection 55

S

- scripts, for system labels 18
- set up connection details 14
- set up content routing 33



- set up contextual routing 30
- set up printer hierarchy 36
- set up specific printer 35
- setup flowchart 12
- specify routing 36
- style sheets. *See* templates
- syntax creation 7
- syntax for label template, label template syntax 7
- syntax types for label printers 14
- system tokens 8
- system tokens, label printing 8

T

- TCP direct connection 7
- TCP HTTP 7
- template syntax examples 7
- templates for DO labels 90
- templates, label print 6
- tokens, define values 8

- tokens, entering in Label Format Maintenance 24

V

- vendors, specify in routing 31
- view and delete datasets 26
- view generating labels 58
- view label print status 53
- view queued labels 57

W

- wildcards, in setup 31
- workfile, label event 45, 65

X

- XML 6

Z

- ZPL 6

