

DYMO Developer SDK Support Blog

FOR DEVELOPERS USING DYMO SOFTWARE DEVELOPMENT KITS

Search

Nov
17
2021

Driver Only Package

Uncategorized

No Responses »

RSS Links

Recent Posts

Recent Comments

Categories

DYMO Label Framework

Featured Software

News

Polls

SDK News

SDK Samples

Tips

Uncategorized

Tags

.NET 5.1 8.3.1 8.5.3 10.7 11 80040154
Address Android barcode browser C#
Chrome COM CUPS deployment DLS
SDK docs documentation DYMO
DymoAddin DYMO Label error FAQ file
Firefox Formatting
Framework google How-
To IE9 iOS JavaScript js
label LabelSet LabelWriter Lion
printing QR Safari SDK VB.NET
web xml

Recent Posts

Driver Only Package
DYMO Connect SDK
Updated JS SDK and DLS
Upcoming SDK Web Service Changes
Recent issues with slow printing

The Beta version of Label Writer 5xx Driver only package is ready.

This release adds support for the Label Writer 5xx series printers, The beta version of Driver can be download from here

Please check the DYMO Technical Reference document for how driver is working.

About the LabelWriter 550 Series Printers

The LabelWriter 550 series printers (LabelWriter 550, 550 Turbo, 5XL) are high-performance, low-cost printers used for printing mailing labels, postage, file folder labels, bar code labels, and more. The LabelWriter 550 and LabelWriter 550 Turbo printers have a 57 mm wide print head. The LabelWriter 5XL has a 101 mm wide print head. All printers have 300-dpi print head.

The 57 mm wide thermal print head uses 672 individually addressable dots to form individual raster lines of data at 300 dots per inch across the print head—the 101 mm wide print head uses 1248 dots. Both use 300 dots per inch in the travel direction as directed by the print control data. Because they use specially treated, heat-activated paper, the printers require no ink, toner, or other refills.

All 550 series printer models connect to a host computer through a standard full-speed USB 2.0-compatible interface. There are no built-in fonts. The host computer is responsible for sending commands and data to the printer to form each individual raster line of data. This is generally performed by printer drivers in the host computer that convert the image of the label into the proper command and data stream required by the printers.

LabelWriter 550 Turbo and 5XL printers can connect to a host through a 10M/100M Ethernet LAN connection.

Print Data

Print data is encoded in strings of 8-bit bytes.

The default value for LabelWriter 550 and LabelWriter 550 Turbo printers is 84 bytes per line (84 * 8 = 672, the number of dots in the print head)—for LabelWriter 5XL, 156 bytes (156*8=1248, the number of dots in the print head). The host-computer can send fewer bytes than the maximum within the <esc> D command. The control electronics do not check the validity of the new value. Therefore, it is the responsibility of the host-software to send only those values which are valid for the width of the print head.

The Bytes-Per-Line variable is held by the control electronics until they are changed by a new command sequence, is reset to default values by a power-on reset, a software reset command, or after the end of a print job.

Print Job Structure

The print job structure consists of the following parts:

1. Print Job Header
2. Label Structure
 - a. Label Header
 - a. Label Data
 - a. Label Trailer
3. Print Job Trailer

Print Job Header

The Print Job Header is the beginning of a print job structure. It is mandatory to have <ESC s> to let the printer know the start of the print job.

The table below shows the mandatory command for the header section along with optional commands that can be placed in this section.

Mandatory Commands	
Start of Print Job	<ESC s>
Optional Commands	
Set Maximum Label Length	<ESC L>
Select Text Output Mode or Select Graphics Output Mode	<ESC h> or <ESC i>
Select Energy Setting	<ESC C>
Set Output Tray * (will be supported by LW550 Twin Turbo)	<ESC q>
Select Content Type	<ESC T>

Label Structure

The Label structure has the following 3 parts:

1. Label Header
2. Label Data
3. Label Trailer

The structure must be present for each label that is to be printed in a single print job.

Label Header

The Label Header is the beginning of the label structure. It is mandatory to have <ESC n> to indicate the label index and <ESC D> to indicate the start of the label print data. See the “Printer Commands” later in this document for more details.

The table below shows the mandatory command for the header section along with optional commands that can be placed in this section.

Mandatory Commands	
Set Label Index	<ESC n>
Start of Label Print Data & Label Print Data	<ESC D>

Label Data

The Label Data is the data that will be printed out onto the labels. It is an array of binary data bytes. It's size equals width*height. '0' means a dot is not printed; '1' means a dot is printed. The most significant bit of the first print data byte represents the lower pixel of the first line. See the <ESC D> command in the “Printer Commands” section later in this document for more details.

Bytes per line= **roundup**((height * Bytes Per Pixel) / 8)

Label Trailer

The Label Trailer is end of the label structure. It moves the label to either the print head of the next label or to the tear position. It is mandatory that <ESC G> (Feed to Print Head) or <ESC E> (Feed to Tear Position) be used in this section.

Mandatory Commands	
Feed to Print Head	<ESC G>
Feed to Tear Position	<ESC E>

Print Job Trailer

The print Job Trailer is the final part of the print job structure. It is mandatory to use the <ESC Q> command to end the print job.

Data Synchronization

After the printer receives a Lock Request from the host, the printer will respond. (See Print Status Response section for details.) The host (Host-A) will be granted a lock only if the printer is not printing. Once the lock is granted, the Host-A will begin to send the print data for the current print job with the <ESC s> command. (See “Print Job Structure” section above for additional details). If another host (Host-B) requests the printer’s status, the printer will respond that it is printing. No other host can interrupt the print job of Host-A.

The data goes into a buffer and the printer sorts through the various commands applying the print job settings.

The data between the height value in the <ESC D> command and before <ESC G> or <ESC E> is the data that will be printed, also referred to as print data.

Label Positioning

A motor-driven platen, which acts as a pinch roller, controls label movement within the printer. The control electronics track label travelling positions. An infrared LED photocell detects the top-of-form sense hole that is located between labels. The absolute positions of the label and the tear bar are calculated based up the reading of an infrared LED photocell sensor.

Top-of-Form Sync

At power up, label position is always assumed to be correct. If an Out of Paper condition is detected at power up, user requires to press form feed button and the stepper motor will advance until label load to Top-of-Form tear bar position.

Label Length

The label length is determined by the SKU data found on the NFC Tag.

If the roll has continuous form paper the Form Feed command (<esc> E) is changed to feed enough dot lines to allow for the last line of print data to extend past the printer tear-bar.

The host can use <esc> L to Set Length to Continuous Stock. See more details in the “Printer Commands” section later in this document. Only authentic Dymo labels with a valid NFC Tag can be used for printing.

Label-Movement Commands

There are two label-movement commands implemented in the LabelWriter 550 Series: Form Feed, Short Form Feed. These commands operate as follows.

Form Feed Command (<esc> E)

This command advances the most recently printed label to a position where it can be torn off. This positioning places the next label beyond the starting print position. Therefore, a reverse-feed will be automatically invoked when printing the next label. Use Feed to Print Head (Short Form Feed) to optimize print speed and to eliminate the reverse feeding when printing multiple labels.

Short Form Feed Command (<esc> G)

This command feeds the next label into print position (die-cut labels). The most recently printed label might still be partially inside the printer and cannot be torn off or cut. This command is meant to be used between labels on a multiple label print job.

Use this command when printing multiple labels. The normal Form Feed command also works for this purpose. However, using the Short Form Feed command optimizes the printer and minimizes print time by eliminating the need to reverse feed after advancing the top-of-form mark to the tear bar, as the normal Form Feed command does. The Short Form Feed command advances the printed label just far enough to place the next label in position for printing. Because of the mechanics of the printer, the preceding label will still be partially within the label exit slot. For a label to be in tear-off position, you must use the normal Form Feed command (<esc> E).

Print Status Response

The LabelWriter 550 series responds to status request with 32 bytes dedicated to reporting its current state.

The host requests the status from the printer by issuing an <ESC A> command (See the “Printer Command” section later in the document for more details).

Self-Test Capability

The printers include a self-test capability in the form of canned print patterns. To activate the self-test, press the form-feed button and power button together and hold it down for approximately 10 seconds. The self-test will start printing a repeating series of test patterns. To stop the self-test, press the form-feed button a second time.

Print Mode

The LabelWriter 550 series features the following print modes:

- Text
- Barcode and Graphics

The LabelWriter 550 and LabelWriter 550 Turbo has the additional print mode:

- High Speed ***NOTE: Not all label rolls have the high-speed feature***

Printer Model	Print Mode		
Text	Barcode and Graphics	High Speed	
DYMO LabelWriter 550	ü	ü	ü
DYMO LabelWriter 550 Turbo	ü	ü	ü
DYMO LabelWriter 5XL	ü	ü	N/A

For better accuracy of dot placement and sizing, the LabelWriter 550 series printers incorporate a feature which allows the host computer to specify a 300 x 300 dpi Barcode and Graphics print mode. In Barcode and Graphics mode, the print quality is improved by a reduction in print speed which provides smooth and even printing, resulting in greater regularity of dot darkness and size. This mode may improve print quality of smaller barcodes and some graphic images. This creates more precise dots for better image quality.

Low Power Mode

The LabelWriter 550 series includes an automatic power saving feature. After 30 seconds of inactivity the printer will shut down its unused internal peripherals and dim its LEDs. Pressing the power button while the printer is in this state will cause it to resume normal operation rather than feeding a label. If print data is sent to the printer while it is in Low Power mode it will automatically resume full operation.

Power Requirements

The LabelWriter 550 series printers are normally supplied with an external, universal (100 – 240 VAC) AC power adapter. For OEM applications, a DC voltage can also be supplied to the main printed circuit board (PCB) of the printer using the JP2 connector.

Note: The printers have been tested and certified for EMC and safety requirements using the supplied AC adapter. Supplying a non-DYMO power source may change the EMC and safety certifications.

Input power is variable based on the amount of information printed and the duty cycle of the label printing.

The printer driver (print commands from the host computer) controls some printing parameters and will change maximum power usage.

AC Adapter

LabelWriter 550	
Model	DSA-42PFC-24 2 240175 (42.0W)
Input voltage	100 – 240 VAC, 50/60 Hz, 1.2 Amps
Output voltage	24 VDC, 1.75Amps
LabelWriter 550 Turbo	
DYMO part number	DYS865-240250W (60.0W)
Input voltage	100 – 240 VAC, 50/60 Hz, 1.5 Amps
Output voltage	24 VDC, 2.5Amps
LabelWriter 5XL	

DYMO part number	DSA-96PFB-24 2 240375 (90.0W)
Input voltage	100 – 240 VAC, 50/60 Hz, 1.5 Amps
Output voltage	24 VDC, 3.75Amps

Connectors

JP2:

DC jack, 5.5 mm O.D. x 2.5 mm I.D. Center Positive

USB Printer Interface

The LabelWriter 550 series printers all communicate with the host computer using a full-speed USB 2.0 interface. This interface also operates with USB Version 1.1 or later. The printers implement the standard USB Printer Class Device interface for communications (see <http://www.usb.org/developers/devclass/>).

While it is not the purpose of this section to define how to write a USB driver for a printer device, this section provides some product specific information that is required for any USB device.

Each USB device provides plug-and-play information when it is connected to a host computer. This information includes a manufacturer specific, or Vendor, ID (VID) and a product specific ID (PID). These identifiers, when combined with the class type provided during USB enumeration, are sufficient to identify the driver to load.

USB Vendor ID

The USB Vendor ID (VID) assigned to DYMO by the USB Implementers Forum is 0x0922h (2338 Decimal).

USB Product IDs

Each printer has a unique Product ID, as shown in the table below.

Printer Model	Product ID (Hex)
DYMO LabelWriter 550	0x0028
DYMO LabelWriter 550 Turbo	0x0029
DYMO LabelWriter 5XL	0x002A

LAN Printer Interface

The LabelWriter 550 Turbo and LabelWriter 5XL printers are also able to communicate with the host computer using 10M/100M Ethernet LAN connection.

The printer will auto-connect upon ethernet cable plugin into the RJ45 connector.

IP Address

IPv4 protocol is supported.

By default, IP address of the LabelWriter 550 Turbo and LabelWriter 5XL printers will be obtained by DHCP. Communication will be enabled upon successful assignment of IP address.

mDNS

The mDNS protocol is used to discover printer on the network and conforms with mDNS protocol (RFC 6762/6763) specifications. The port used for this service is 5353.

Below are the mDNS strings used for printers with a LAN interface:

Printer	Supported Profiles & protocols
DYMO LabelWriter 550 Turbo	DYMO LabelWriter 550 Turbo on DYMOLW550T<6 digit MAC Address>E
DYMO LabelWriter 5XL	DYMO LabelWriter 5XL on DYMOLW5XL<6 digit MAC Address>E

Printer Commands

All LabelWriter 550 series printers implement the same basic commands.

Commands Sent to the Printer

The commands implemented by the LabelWriter printer are as follows. Note that each command is listed with its mnemonic (<ESC> s as well as its hexadecimal representation (1B 73). Parameter values are specified as "n" for single byte values or "n1 n2" for two-byte values.

ESC s Start of Print Job

1B 73

Indicates a new print job. A unique job ID is passed along with this command.

Format:

BYTE	0	1	2	3	4	5
	'ESC'	's'	Job ID			

ESC L Set Maximum Label Length

1B 4C

Sets the print engine mode between normal label stock and continuous label stock. Normal label stock is the default mode for the print engine.

ESC h Select Text Output Mode

1B 68

Tells the print engine that the label shall be printed with the print settings which are ideal for text. Text mode is the default printer setting.

ESC i Select Graphics Output Mode

1B 69

Tells the print engine that the label shall be printed with the print settings which are ideal for graphics and barcodes. The print speed might be reduced in this mode.

ESC T Content Type

BYTE	0	1	2
	'ESC'	'T'	Speed Mode

1B 74

Sets the speed of the printer. Default is Normal Speed (0x10).

Byte Value	Meaning
0x10	Normal Speed
0x20	High Speed

ESC n Set Label Index

1B 6E

Sets the label index. The same label index is returned in the print status providing the host the possibility to track which label is being printed.
Default value is 0.

BYTE	0	1	2	3
	'ESC'	'n'	Label Index	

ESC D Start of Label Print Data & Label Print Data

1B 44

Indicates the start of the label print data and notifies the print engine about the label's height and width. The height is defined in dots while the width specifies the number of lines. The width does not include the leader and trailer.

Format:

BYTE	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	...	n
	'ESC'	'D'	BPP	Align	Width	Height	Print Data										

Byte	Name	Description
2	BPP	Default value is 1.
3	Alignment	Print Alignment 2: Bottom
Byte 4 to Byte 7	Width	Number of lines Range: 0 to 2 ³²
Byte 8 to Byte 11	Height	Number of dots Range: 0 to 2 ³²
Byte 12 to Byte n	Print Data	The data that will be printed onto the label(s)

ESC G Feed to Print Head (Short Form Feed)

1B 47

Feeds the next label into print position. The most recently printed label might still be partially inside the printer and cannot be torn off. This command is meant to be used between labels on a multiple label print job.

ESC E Feed to Tear Position (Long Form Feed)

1B 45

Advances the most recently printed label to a position where it can be torn off by the automatic cutter. This positioning places the next label beyond the starting print position. Reverse-feed will be automatically invoked when printing the next label.

ESC Q End of Print Job

1B 51

Indicates the end of a print job. Upon reception of this command the print engine will release the connection and start accepting other print jobs.

ESC A Request Print Engine Status

1B 41

Requests the current printer status, such as errors and label size of inserted label printers. It takes a lock byte as parameter. When this lock byte contains a value other than zero, the print engine will attempt to reserve the interface for printing for the maximum duration of 10 seconds or until useful commands are received (any print jobs from other sources are rejected during this period). If useful data activity lapses for 10 seconds or more at any point during the connection, the connection shall be closed by the print server. Printer driver uses ESC A 0, 1 and 2 to query printer's status.

Print Status Response:

Byte	Bits	Name	Description
Byte 0	b7...b0	Print status	The actual print engine status. Type: predefined values Default value: 0 Range: 0 = idle:

1 = printing
 2 = error 3 = cancel 4 = busy 5 = Unlock 0-3: Once lock is granted to active host, status replies by active host 4: after printer wakes up from standby 5: Status reply before lock is granted to active host

The Job ID of the ongoing print process.

Byte 4... Byte 1	b31...b0	Print job ID	Type: u32 Default value: 0 Range: 0 = printer idle 1...2 ³² = Job ID
---------------------	----------	--------------	---

The index of the label/page currently being printed.

Byte 6... Byte 5	b15...b0	Label index	Type: u16 Default value: 0 Range: 0...2 ¹⁶ = index
---------------------	----------	-------------	--

Byte 7	b7...b0	Reserved	Default value: 0
--------	---------	----------	------------------

The actual thermal print head status.

Byte 8	b7...b0	Print head status	Type: predefined values Default value: 2 Range: 0 = ok 1 = overheated 2 = status unknown
--------	---------	-------------------	---

The actual print density setting in %.

Byte 9	b7...b0	Print density	Type: u8 Default value: 100 Range: 0 = disable printing 1...200 = density in %
--------	---------	---------------	--

The status of the main bay.

Byte 10	b7...b0	Main bay status	Type: predefined values Default value: 0 Range: 0 = bay status unknown 1 = bay open; media presence unknown 2 = no media present 3 = media not inserted properly 4 = media present – media status unknown 5 = media present – empty 6 = media present – critically low 7 = media present – low 8 = media present – ok 9 = media present – jammed 10 = media present – counterfeit media
---------	---------	-----------------	--

Byte 22... Byte 11	B95...b0	SKU Info	The SKU of the inserted consumable Type: char Default value: 0 (empty) XXXXXXXXX = SKU
-----------------------	----------	----------	---

The ID of the present error.

Byte 26... Byte 23	b31...b0	Error ID	Type: u32 Default value: 0 Range: 0 = no error present 1...2 ³² = error codes
-----------------------	----------	----------	--

Byte	B15...b0	Label	Remaining count of inserted consumable Type: u16 Default value: 0 (empty)
------	----------	-------	---

28		count	
Byte			
27			
Byte	b3...b0	EPS Status	The status of the external power supply. 1 = EPS present
b7...b4	Reserved	Default value: 0	
			Print Head Voltage
Byte	b3...b0	Print Head Voltage	Type: predefined values Default value: 0 Range: 0 = unknown 1 = ok 2 = low 3 = critically low 4 = too low for printing
b7...b4	Reserved	Default value: 0	
Byte	b7...b0	Reserved	Default value: -1
31			

ESC C Set Print Density

1B 43

Sets the strobe time of the printer to a given percentage of its standard duty cycle. A lower value results in lighter printouts while a higher value leads to darker printouts. Default value is 100. Duty range is 0-200%.

BYTE	0	1	2
	'ESC'	'C'	Duty

ESC e Reset Print Density to Default

1B 65

Resets the print density to default value of 100%.

ESC U Get SKU Information

1B 55

Used to retrieve the inserted LW550 Consumable SKU information from NFC.

The following is the 63-Byte response to ESC U:

Byte	Bits	Name	Description
Byte			
1...	b15...	Magic	0xCAB6
Byte	b0	number	
0			
Byte	b7...b0	Version	Current version is '0'
2			
Byte	b7...b0	Length	
3			
Byte			
7...	b15...	CRC	
Byte	b0		
4			

Byte 19...	b95... b0	SKU #	The SKU # of inserted Type: char Default value: 0 Range: 1...12 = SKU #
			Type: predefined values
Byte 20	b7...b0	Brand ID	Default value: 0x00 0x00 = DYMO 0x01-0xFF = Undefined
			Type: predefined values
Byte 21	b7...b0	Region	Default value: 0xFF 0xFF = Global 0x00-0xFE = Undefined
			The type of label material
			Type: predefined values
Byte 22	b7...b0	Material Type	Range: 0x00 = Card 0x01 = Clear 0x02 = Durable 0x03 = Paper 0x04 = Permanent 0x05 = Plastic 0x06 = Removable 0x07 = Time exp. 0x08 – 0xFF = Undefined
			Label Type Type: predefined values
Byte 23	b7...b0	Label Type	Range: 0x00 = Continuous 0x01 = Die 0x02 = Card 0x03 – 0xFF = Undefined
			Label Color Type: predefined values
Byte 24	b7...b0	Label Color	Range: 0x00 = Clear 0x01 = White 0x02 = Pink 0x03 = Yellow 0x04 = Green 0x05 = Blue 0x06 – 0xFF = Undefined
			Content Color Type: predefined values
Byte 25	b7...b0	Content Color	Range: 0x00 = Black 0x01 = Red/Black 0x02 – 0xFF = Undefined
			Marker Type Type: predefined values
			Range: 0x00 = Marker 1 front edge indicates offset to cut location and offset to start of label 0x01 = Marker 1 front edge indicates offset to cut location and Marker 1 rear edge indicates offset to start of label 0x02 = Marker 1 front edge indicates offset to start of label and Marker 1 rear edge indicates offset to cut location 0x03 = Marker 1 front edge indicates cut location and Marker 2 front edge indicates offset to start of label 0x04 – 0xFF = Undefined
Byte 27	b7...b0	Reserved	
Byte 29 ... Byte 28	b15... b0	Marker Pitch	Marker Pitch Type: u16 1 – 2^16 = length in mm
Byte 31 ... Byte 30	b15... b0	Marker1 width	Marker1 width Type: u16 1 – 2^16 = length in mm
Byte 33 ... Byte 32	b15... b0	Marker1 to start of label	Marker1 to start of label Type: u16 1 – 2^16 = length in mm

Byte 35 ... 34	b15... b0	Marker2 width	Marker2 width Type: u16 1 – 2^16 = length in mm
Byte 37 ... 36	b15... b0	Marker2 offset	Marker2 offset Type: u16 1 – 2^16 = length in mm
Byte 39 ... 38	b15... b0	Vertical offset	Vertical offset Type: u16 1 – 2^16 = length in mm
Byte 41 ... 40	b15... b0	Label length	Label length Type: u16 1 – 2^16 = length in mm
Byte 43 ... 42	b15... b0	Label width	Label width Type: u16 1 – 2^16 = length in mm
Byte 45 ... 44	b15... b0	Printable area horizontal offset	Printable area horizontal offset Type: u16 1 – 2^16 = length in mm
Byte 47 ... 46	b15... b0	Printable area vertical offset	Printable area vertical offset Type: u16 1 – 2^16 = length in mm
Byte 49 ... 48	b15... b0	Liner width Liner width	Liner width Type: u16 1 – 2^16 = length in mm
Byte 51 ... 50	b15... b0	Total label count	
Byte 53... 52	b15... b0	Total Length	Length of roll Type: u16 1 – 2^16 = length in mm
Byte 55 ... 54	b15... b0	Counter Margin	Used to determine remaining labels on roll or limit usage
Byte 56	b7...b0	Counter Strategy	Type: predefined values Range: 0x00 = Counting up from 0x0000 to “amount of labels” + “counter margin” 0x01 = Counting up from 0xFFFF – “amount of labels” – “Counter margin” to 0xFFFF 0x02 – 0xFF = Undefined
Byte 59, 57	b23... b0	Reserved	
Byte 61, 60	b15... b0	Production date	DDYY format
Byte	b15...	Production	HHMM format

63, b0 time
Byte
62

ESC V Request Print Engine Version

1B 56

Used to retrieve version information of the print engine.

Format:

BYTE	0	...	15	16	...	31	32	33
	HW Version					FW Version		Product ID

HW Version:

Byte 0 to 15

This is the hardware version as a 16-character UTF-8 encoded string.

FW Version:

Byte 16 to 31

This is the firmware version as a 16-character UTF-8 encoded string.

Byte	Meaning
16 to 19	Application FW Version: "FWAP" Boot Loader FW Version: "FWBL"
20 to 23	Major Release Version
24 to 27	Minor Release Version
28 to 31	Release Date Format: MMY Y

PID:

Byte 32 to 33

Two bytes of USB PID

ESC @ Restart Print Engine

1B 40

Reboots the print engine.

ESC * Restore Print Engine Factory Settings

1B 24

Restores all the factory settings of the printer.

ESC o Set Label Count

Sets label count.

BYTE	0	1	2
	'ESC'	'o'	Count

Posted by Sung Yun at 15:15

Hi everyone,

A few months ago **DYMO Connect Software v1.2**(DCD) was released. This update includes new sharing render functionality in the SDK similar to labels created from DYMO Label Software(DLS) being compatible with DYMO Connect. *DYMO.Connect.SDK* is distributed via Nuget Package and *DYMO.Connect.Framework* via JavaScript.

DYMO.Connect.Framework **needs DYMO Connect v1.2 or later** to be able to open, update, and print labels created from DYMO Connect. It has the same functions of the old DYMO.Label.Framework; therefore, it works with DLS installed as well with an exception. See the follow scenarios:

DYMO.Connect.Framework works

- Opening a label created from DLS with DLS Service running.
- Opening a label created from DLS with DCD Service running.
- Opening a label created from DCD with DCD Service running.

DYMO.Connect.Framework does not work

- Opening a label created from DCD with DLS Service running.

Distribution and samples

DYMO.Connect.SDK

- <https://www.nuget.org/packages/DYMO.Connect.SDK/>

DYMO.Connect.Framework

- <https://github.com/dymosoftware/dymo-connect-framework>

Samples

- <https://github.com/dymosoftware/DCD-SDK-Sample>

DYMO Connect Software

- <https://www.dymo.com/en-US/online-support/dymo-user-guides>

Posted by Marco Lerma at 10:27

May
29
2018

Updated JS SDK and DLS

Uncategorized

283 Responses »

Hi everyone,

Today, we are releasing an updated version of our JS SDK and DLS to address some security and TLS issues. These updates are now live on dymo.com and labelwriter.com. You will find download links below. Please note that the latest version of DLS **requires** the latest version of JS SDK. If you plan to update your users' DLS version then you will also need to update the JS SDK version used in your application.

DYMO Label Framework JS SDK – <http://labelwriter.com/software/dls/sdk/js/DYMO.Label.Framework.3.0.js>

DLS Windows 8.7.2 – <http://download.dymo.com/dymo/Software/Win/DLS8Setup.8.7.2.exe>

DLS Mac 8.7.2 – <http://download.dymo.com/dymo/Software/Mac/DLS8Setup.8.7.2.dmg>

Posted by dymodev at 07:48

May
09
2018

Upcoming SDK Web Service Changes

Uncategorized

20 Responses »

We wanted to take time to highlight some changes that will be coming to the SDK web service in the next update of DLS. The changes are centered around reducing potential security risks as well as improving the overall end-user experience.

- **Behavior change for disconnected users**

Currently, when the JavaScript SDK tries to discover the DYMO SDK web service instance running on the machine, it does so by scanning all ports within a range. The first port that responds is the one that is used. If the JavaScript SDK connects to a web service instance that belongs to a disconnected user, printing will fail.

In the upcoming version of DLS, the web service for each user will suspend itself if the user disconnects (either due to fast-user switching or temporary RDP session disconnect on a remote machine). Once the user becomes active again, the service will resume. This will ensure that only one instance of the web service is active at any given time (*NOTE: for Windows Terminal Server multiple users can be active simultaneously so multiple web service instances can be running*) and will prevent any issues due to multiple web service instances.

The DYMO SDK web service scans a 10 port range in order to find an available port to run on. In the event of all 10 ports being taken (meaning there are 10 active users logged onto a machine), any new users that log on will not have their own web service instance but instead use one of the existing 10 instances. For most common usage scenarios this will not be a problem. However, if the JavaScript SDK connects to another user's instance (due to being the first to respond to the port scanning), shared printers for the other user will be visible through the SDK but shared printers for the current user will not.

- **Random certificate generation**

Root and exchange certificates will be generated randomly using strong cryptographic keys during DLS installation and discarded immediately after installation. This ensures the certificates are unique for every DLS installation and adds an extra layer of security to the DYMO SDK web service.

- **Change in web service URI**

The DYMO SDK web service will be changed to use **0.0.1** instead of **localhost** as its URI to avoid host name spoofing. This means that the existing JavaScript SDK will not work with the new web service installed with DLS. An upgrade to the new version of the JavaScript SDK will be required if a user updates their DLS install. However, the new version of the JavaScript SDK will be backwards compatible with older versions of DLS.

- **Firefox changes**

The new web service for DLS Windows does not install a root certificate into Firefox's local keychain. Instead, it enables Firefox's built-in feature which allows Firefox to use OS certificates. This flag is set on DLS installation and removed on DLS uninstallation.

Posted by dymodev at 13:19

Apr
24
2018

Recent issues with slow printing

Uncategorized

83 Responses »

UPDATE

A new version of DLS is available that resolves this issue. This will need to be installed on any client machine experiencing the slow printing issue. You can download it here:

<http://download.dymo.com/dymo/Software/Win/DLS8Setup.8.7.1.exe>

UPDATE 2

The above version of DLS is now live on DYMO.com for all regions.

<http://www.dymo.com/en-US/online-support/dymo-user-guides>

Since 4/23/2018 all users of the windows version of our SDK have been experiencing a 10-15 second slowdown. The issue can also cause performance issues when opening label files in DLS. This appears to be caused by our code attempting to contact 128.30.52.100 (<http://www.w3.org/1998/XMLSchema>) in order to validate each label before printing.

We have identified a fix for this issue and we are currently expediting a test cycle so we can release a patch as soon as possible.

Until the actual fix goes out, users can alleviate this slowdown by preventing this call. We have seen success with two methods thus far.

1. Prevent connections to 128.30.52.100 (<http://www.w3.org/1998/XMLSchema>)
2. Use the windows defender firewall to prevent DYMO.DLS.Printing.Host.exe from making outbound connections.

We apologize for the inconvenience, and will update this post as soon as the situation has been resolved.

-Regards,

DYMO Team

Posted by dymodev at 15:32

Dec
12
2017

ERR_SSL_VERSION_INTERFERENCE in Chrome 63 using the JS SDK

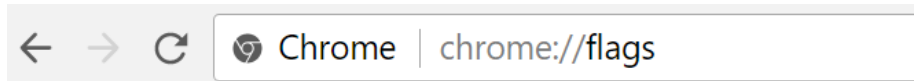
Uncategorized

65 Responses »

Hello everyone,

Chrome recently released an update to Chrome (version 63) that includes experimental support for TLS 1.3. Unfortunately, this is causing problems with our client-side web service that powers the JavaScript SDK. We are currently looking into this issue but in the meantime, there is a workaround available. The following steps need to be performed on affected versions of Chrome:

1. In the Chrome URL bar, enter `chrome://flags` and hit Enter



2. Once on the `chrome://flags` page, find the setting for **TLS 1.3**

TLS 1.3

Sets the TLS 1.3 variant used. – Mac, Windows, Linux, Chrome OS, Android
[#tls13-variant](#)

Default ▼

3. Change this setting to **Disabled**



TLS 1.3

Sets the TLS 1.3 variant used. – Mac, Windows, Linux, Chrome OS, Android
[#tls13-variant](#)

Disabled ▼

4. Relaunch Chrome

RELAUNCH NOW

Once Chrome restarts, you should no longer receive the ERR_SSL_VERSION_INTERFERENCE error. We will update this post as we continue to research this issue.

Posted by dymodev at 16:10

Jun
13
2017

New DLS update released! (This fixes issues on Mac OS X)

Uncategorized

23 Responses »

The latest release of DLS is now available for customers that have been experiencing Chrome security issues with the previous release!

You can download the Mac OS X release via the following link:

Mac OS X Download link:

<http://labelwriter.com/software/dls/mac/DLS8Setup.8.6.2.134.dmg>

Posted by dymodev at 10:29

May
30
2017

New DLS 8 released! (This fixes security certificate issue for Windows)

Uncategorized

170 Responses »

The latest release of DLS is now available for customers that have been experiencing Chrome security issues with the previous release!

You can download the Windows release via the following link:

Windows Download link:

<http://labelwriter.com/software/dls/win/DLS8Setup.8.6.2.658.exe>

The Mac OS X version will follow within the next few days.

Posted by dymodev at 09:44

May
04
2017

Google Chrome “not secure warning” fix is on the way!

Uncategorized

55 Responses »

It has come to our attention that Google Chrome is flagging calls to the DYMO Web Service as “not secure.” This has been causing many of our users to have issues when printing through Chrome.

We have found the issue and have an internal fix undergoing QA testing at this very moment. We estimate the fix will be released within 2 to 3 weeks. We apologize for the inconvenience at hope this will be resolved soon.

In the meantime, please consider several temporary fixes that have been identified within the below blog post:

The new DLS 8.6.1 release is now available!

The latest release of DLS is now available for customers that have been experiencing issues with the previous 8.6 release. Issues fixed in 8.6.1 include: Print Quality issues with a variety of consumables, including File Folder Ability to not install the Web Service and other installer fixes Various SDK fixes (DLLPrinting.dll, RenderLabel requests failing, threading ... Continue reading



DYMO Developer SDK Support Blog

219

Posted by dymodev at 10:57

Mar
08
2017

The new DLS 8.6.1 release is now available!

DYMO Label Framework, Featured Software, News, SDK News

219 Responses »

The latest release of DLS is now available for customers that have been experiencing issues with the previous 8.6 release.

Issues fixed in 8.6.1 include:

- Print Quality issues with a variety of consumables, including File Folder
- Ability to not install the Web Service and other installer fixes
- Various SDK fixes (DLLPrinting.dll, RenderLabel requests failing, threading issues, etc.)
- Various bug fixes

Please use the below links to download the DLS installer.

Mac OS Download link Mac DLS 8.6.1

Windows Download link Win DLS 8.6.1

Posted by dymodev at 11:45

[Older Entries](#)