Loftware Print Server Version 12.1 Release Notes
April 2020

What new features are available in this release?

Failover with an external proxy: A Health Application port has been added to LPS to enable failover between identical LPS servers with the use of a third-party load balancer.

Manage color using label scripting: Label scripting now supports the ability to access and modify foreground color of text and shapes.

Support for Windows Server 2019: LPS/LLM can now be implemented on servers running Windows Server 2019.

Removal of old scripting engine: In LPS/LLM 12.0, support for the Microsoft Chakracore scripting engine was added, with the option to continue to use the old Spider Monkey scripting engine. The new Microsoft Chakracore scripting engine offers significant performance improvements when printing labels with scripts. In this release, the old Spider Monkey scripting engine has been removed and completely replaced by the new Microsoft Chakracore scripting engine. The old Spider Monkey scripting engine is no longer available in this release of LPS/LLM.

> **Important!** Prior to upgrading to LPS/LLM 12.1, back up LPS/LLM.

What fixes are available in this release?

Barcode related fixes:

- Resolved an issue to correctly print Datamatrix barcodes to Zebra printers.
- Resolved an issue with printing Datamatrix barcodes with data that contains underscores on the Zebra Plus Series of printers.
- Resolved an issue with custom Inc/Dec functionality when printing Code 128 barcodes on a Zebra printer and using a field-level override that is managed in Preferences under Zebra > IncrementDecrement.
- Resolved an issue to enable a greater than (>) symbol to be encoded in a Code 128 barcode when printing to a Zebra printer.
- Resolved an issue to enable a tilde (~) symbol to be encoded in a Code 128 barcode.
- Resolved an issue that was resulting in QR Code barcodes being printed as black squares when printing through the On Demand Print.
- Resolved an issue in which labels containing a QR Code with the Symbol Size set to a specific size rather than to auto fail to print.

Performance related fixes:

- Resolved an issue that was negatively impacting the performance of the IPL Family Driver when printing TrueType® fonts.
- Resolved an issue that was limiting performance logging to 499 printers.
- Resolved an issue that was causing the LPS service to hang upon shutdown and restart.

Printing related fixes:

- Resolved a software crash that occurred when printing TIF images with full paths that are larger than 265 characters.

- Resolved a software crash that occurred due to sending special characters in XML files.

- Resolved an issue that caused an intermittent error in jobs submitted via TCP/IP that include special characters such as an ampersand (&) or the less than (<) symbol.

- Resolved an issue that caused inverted text fields containing a TrueType® font to be printed with the text not inverted.

- Resolved an issue that transposed the height and width of images having a transparent background, potentially truncating the images.

- Resolved an issue with font spacing when using a Text Box with dynamic point size, horizontal scaling, and 270-degree rotation.

- Resolved an issue in which canceling an actively printing job in the Loftware Status Dialog window caused the next job to remain pending indefinitely unless the user made a change to the device configuration.

Script related fixes:

- Resolved an issue in which the Microsoft Chakracore scripting engine did not support the inclusion of CRLF in an XPath Query and failed the script by presenting an Expected Identifier error.

- Resolved a software crash that occurred when a percent (%) symbol was sent in the job data in some situations.

- Resolved an issue with label scripts that was causing jobs to fail with the following error message: End tag was not expected at this location.

- Resolved an issue with label scripts failing to report the line number on which an error occurs on when verifying a script.

Other:

- Resolved an issue that was resulting in incomplete WatchDog logs.

- Enhanced Batch Print Debug logging to capture when calls and connections are made to and returned from ODBC.

- Improved logging in cases in which the WDNT Service Account user is changed and does not have appropriate registry key access.

- Resolved an issue that was causing the Delete button to be disabled when a Web User was selected in Device Connection Options.

- Enhanced the LPS installation process so that the option to initialize the print server defaults to "Yes".

What issues are known about this release?

Automatic updating of existing scripts: Saving an existing label with a script in LPS/LLM 12.1 automatically updates that script to be compatible with the new Microsoft Chakracore scripting engine even if the user did not change the script. After being saved, the script is incompatible with LPS/LLM versions prior to LPS/LLM 12.0.2.

Differences between the new Microsoft Chakracore scripting engine and the old Spider Monkey scripting engine:

- The old scripting engine used XPath XML queries, whereas the new scripting engine uses JSON queries and could present an Unexpected Character error.

- The old scripting engine touched every field, whereas the new scripting engine only updates fields that are required per the script.

- The old scripting engine had a property named **length** to describe the maximum length of data, whereas the new scripting engine reserves the property named **length** for an array of fields. Instead, the new scripting engine has a property named **maxChars** to describe the maximum length of data. Note that **maxChars** is in units of bytes rather than characters. If using multi-byte characters, divide this value by the number of bytes per character to determine the maximum number of characters allowed. For example, Simplified Chinese characters require 3 bytes per character, so if **maxChars** is set to 6, then 2 Simplified Chinese characters are allowed.

- The old scripting engine trimmed spaces from every field, whereas the new scripting engine does not do this.

ESCC substitution characters in Datamatrix barcode data: Changes were made to support the printing of Datamatrix barcodes on Zebra printers that relate to FNC1, the underscore (_) character, and the Escape Sequence Control Character (ESCC) command. Zebra's default value for the ESCC was the underscore (_) in old firmware but was changed to the tilde (~) in newer firmware. Until this change, Loftware's ZPL driver for LPS hardcoded the underscore as the ESCC. After this change, the tilde is used as the default ESCC, but if a tilde appears in the user data then another character is substituted.

- In the following situations, the data will not be encoded correctly by default.

    - A Zebra printer with newer firmware is used and the user data contains all of the ESCC substitution candidate characters. The ESCC defaults back to the tilde even though it is not available for use.

    - A Zebra printer with older firmware is used and the user data contains a tilde.

- If the data is not encoded correctly because of one of the preceding situations, the following workarounds are available.

    - If there are no underscores in the data, you can restore the old behavior of using the underscore for both the ESCC and the FieldHex character. This is a system-wide setting and is not configurable in Preferences. To make this change, configure the following setting in LLMWDN32.INI.

      ```
      [Zebra]
      DmtrxUScoreEscc=1
      ```

    - You can configure Datamatrix barcodes to print as a bitmap. This is a printer model specific setting. To make this change, configure the following setting in LLMWDN32.INI.

      ```
      [BarcodeBitmaps]
      datamatrix=~printerId~
      ```

Setting changes that you can revert if necessary:

- Changes were made to correctly encode greater than (>), tilde (~), and caret (^) symbols when they are included in a Code 128 or GS1-128 Generic barcode when printing to a Zebra printer. If you want to restore the behavior that existed prior to these changes, you can do so by configuring the following settings.

  - Previously, the character mask for GS1-128 Generic did not include some symbolic characters that should be supported according to the specification. The greater than symbol was one of these characters.

    The following are all of the characters that are now part of the default character mask for GS1-128 Generic.

    !"#%&'*+,-./0123456789:;<=>?ABCDEFGHIJKLMNOPQRSTUVWXYZ_abcdefghijklmnopqrstuvwxyz()

    If you want to restore the previous behavior, configure the following setting in LLMWDN32.INI . This setting affects all drivers. If this setting is omitted from the INI file, its default value is 1.

    ```
    [Options]
    GS1ISO646forGS1_128GEN=0
    ```

  - Previously, when stepping through the data, the greater than symbol was left as is in the data, regardless of the field type or printer. Now when encountering a greater than symbol in a native Code 128 or GS1-128 Generic barcode on a Zebra (ZPL) printer, the greater than symbol is replaced with character sequence of ">0".

    If you want to restore the previous behavior, configure the following setting in LLMWDN32.INI . Although this setting affects all drivers, Zebra (ZPL) is the only one that substitutes a different character sequence for the symbol. If this setting is omitted from the INI file, its default value is 1.

    ```
    [Options]
    SubstGTinCode128=0
    ```

  - For a Zebra (ZPL) printer, the Code 128 and GS1-128 Generic data string includes embedded switch codes. When building the data string, the presence of a greater than sequence (>0) is processed as a trigger to switch to subset B if necessary. This same setting applies to the tilde (>=) and caret (><) sequences. None of these can be individually configured.

    If you want to restore the previous behavior, configure the following setting in LLMWDN32.INI . This setting affects only Zebra. If this setting is omitted from the INI file, its default value is 1.

    ```
    [Zebra]
    Switch2B4SpecialChar=0
    ```

- Previously, paragraph fields with right justification that were partially off the label were printing left justified. This behavior has been changed so that if the right justification setting is enabled, then the text will always print right justified, regardless of whether the paragraph field extends off the label.

  If you want to restore the previous behavior, configure the following setting in LLMWDN32.INI . If this setting is omitted from the INI file, its default value is 0.

  ```
  [Options]
  ForceOffLabelJustifyToLprnt=1
  ```

Status Client does not automatically fail back: If failover between identical LPS servers is managed by using a third-party proxy and the Status Client is open when Server A goes down, then the Status Client will automatically switch to Server B. However, when Server A comes back online, the Status Client does not automatically switch back. The Status Client must be restarted to switch back to Server A.

End of Support Notice for Windows Server 2008 R2 and Windows Server 2008: New LPS family products no longer support Windows Server 2008 R2 (beginning with LPS 12.1) or Windows Server 2008 (beginning with LPS 11.0). Customers running Loftware solutions on these operating systems are encouraged to upgrade to Windows Server 2019 or other supported operating systems.

End of Support Notice for Windows 7 and Windows Vista: New LPS family products no longer support Windows 7 (beginning with LPS 12.1) or Windows Vista (beginning with LPS 12.0). Customers running Loftware solutions on these operating systems are encouraged to upgrade to Windows 10 or other supported operating systems.