

PrintList Pro

Version 5.3.2

Release notes

September 26, 2016

Published by e-Node worldwide

www.e-node.net/plp

PrintList Pro version 5 is a complete rewrite, yet compatible with legacy 4.x code.

System requirements

PrintList Pro version 5.3 is compatible with 4D v11 to v15, for both MacOS and Windows. It requires MacOS X 10.5 or higher and Windows XP SP2 or better.

4D v15 R releases

PrintList Pro version 5.x is not compatible with 4D's v15 R releases a.k.a. v16 beta in 64 bit developer mode. We are currently working on PrintList Pro version 6, which will support this context and will be made available before 4D v16 goes public.

Check out the news on www.e-node.net and forums.e-node.net.

Warning: behavior change in 5.3

- Since version 5.3, string formatting strips unused characters from the format instead of replacing "number sign" placeholders by non-breaking space (e.g. "1234" formatted with "###-###-###" will produce "1-234", not " - 1-234" as before)

Version 5.3.2 – Fixes

- Break row dividers are honored even when no text is set using *PL_SetBrkText/PL_SetBkHText*
- Fixed a bug using `\\FUNCTION` in breaks (the starting row of the break range could be wrong with large data)
- Fixed possible crash (right-aligned, wrap, leading space, word not fitting into the column)
- Fixed drawing of background color in multistyled text (no "white hair lines" between characters) issue <http://forums.e-node.net/viewtopic.php?f=10&t=286>

Version 5.3.1 hotfix 1

- Fixed header/footer height computation when height is set to variable (second parameter to *PL_SetHeight* set to zero)

Version 5.3.1

- Fixed background color being ignored when a long word is split (background color set in the attributed string)

Version 5.3 – New features

- **Support of the new e-Node Online automatic registration:** this optional registration removes the need to retrieve the serial information or machine ID from the end user setup
- New parameters to *PL_Register* for Online automatic registration:
PL_Register (licenseKey ; options ; email)
 - options is a longint combining up to 4 bits :
 - bit 0: force check (existing) - if this bit is on (true), the provided key is tested regardless of current registration state. If the plugin was not previously registered and the result is 0, it is registered the same way as if the bit was off (or the whole parameter omitted)
 - bit 1: confirm connection "Is it OK to connect to e-Node's license server to register PrintList Pro?"
 - bit 2: display alert if registration error
 - bit 3: display alert if registered(default is no force check, silent mode)
 - email is a text parameter holding your email address if you want to be notified when registration is performed (or fails) including the key that was issued and user IP information
- Support of the **online instant activation:** Master (unserialized) keys can now directly be entered in the demonstration mode dialog. One-click registration, no need to recompile, no need for serial numbers or machine IDs !
- Added support of background color in attributed text (stored internally by 4D as)
- **New behavior:** string formatting strips unused characters from the format instead of replacing "number sign" placeholders by non-breaking space (e.g. "1234" formatted with "###-###-###" will produce "1-234", not " - 1-234" as before)
- MacOS 64 bit server support (not 64 bit developer version – 4D v15Rx)

Version 5.3 – Fixes

- Fixed a bug in calculations (when PrintList Pro needs more than 25000 points for printing, the calculated values were incorrect (reset after reaching the limit — bug introduced in 5.1b8)
- Fixed break processing when the area is configured both during On Load and On Printing Detail
- Fixed text printing when used with 64-bit 4D server on Mac
- Fixed a bug where some headers could be omitted from the printout

Version 5.2.2

- Named formats are supported: "|format" is resolved when it is set (loading from XML with *PL_Restore*, or setting it directly with *PL_SetFormat*)
 - text formats will only access the formats available in the host database, not components

Version 5.2

- New command: *PL_GetVersion*, does not trigger the demo dialog when not registered (allows to check version before registering)
- Fixed empty last page break (multiple empty break footers with page break)
- Fixed a crash in *PL_SetColBackColor*
- Fixed printing in 4D v14: when a record produces no data, the rest of the records was not printed (using the last empty record)
- When multiple page breaks are requested (multiple footers having page break set), only one page break should be used
- 4D v14 specific change: area is fully re-initialized while printing each new (except the first one), because the area was not destroyed/re-created on every record when **PRINT SELECTION** was used in early 4D v14 versions (bug ACI0084258)
- When there is a picture column and *PL_SetFormat* sets "use picture height", *PL_SetHeight* ignores the fourth parameter numRowLines (number of lines in data rows); variable row height is used

PrintList Pro version 5 – Compatibility notes

- PrintList Pro version 5 supports Unicode for printing.
- Styled text is supported.
- PrintList Pro version 4.x commands are still here: your previous code should work fine, give or take the few changes and minor deprecated features below.
- The *PL_Register* command takes just one parameter, and it returns 0 for OK and an integer between 1 and 8 if not OK.

- The 4D project method *Compiler_PLP* is no longer needed.
- In addition, PrintList Pro uses native drawing. Not all fonts contain italic (or bold) variations and those technologies do not synthesize styles. For example Geneva on most Macs has only the Regular typeface, Arial has Regular, Bold, Italic and Bold Italic typefaces. If any styles are not present in the OS, PrintList Pro will print regular (plain) characters.
- The following features are removed:
 - picture escapes in text
 - outline
 - shadow
- The following commands have been removed:
 - *PL_GetAdvProps* (did nothing in PrintList Pro 4.7, anyway)
 - *PL_SetArrays* (must use *PL_SetArraysNam*)
 - *PL_SetHeaderIcon* (use *PL_SetCellIcon* with row = 0)
 - *PL_SetSubSelect*
 - *PL_SaveData* (was already document as obsolete – use the new *PL_Save* in XML format)
 - *PL_RestoreData* (was already document as obsolete – use the new *PL_Load* in XML format)
- *PL_SetCellIcon* now only supports pictures from the Picture Library. 'cicn' and 'PICT' resources are no longer supported.
- Escape characters ("^1234" in cell data or headers) are no longer supported, including the first parameter to *PL_SetMiscOptions*)
- Patterns are no longer supported. They are interpreted by PrintList Pro version 5 as transparency ratios:
 - "black": 100%
 - "darkgray": 75%
 - "gray": 50%
 - "lightgray": 25%
 - "white" or "none" or "" or anything else: 0% = no drawing

PrintList Pro version 5 – New Configuration commands

- *PL_AddColumn* (areaRef:L; dataPointer:Z; insertAt;L) -> result:L
 - add column at position insertAt; zero means append to the end
 - when in Records mode, dataPointer should be a pointer to a field
 - when in Arrays mode, dataPointer should be a pointer to an array (must not be a local array!)
 - this command supports the component architecture (using arrays from the host database in a component and vice versa)

- *PL_Load* (areaRef:L; XML:T) -> result:L
 - initialize an area from XML
 - Note: *PL_Load* can be used without any other command use (e.g. no defined columns - they will be read from the XML). But don't use *PL_Load* more than once - it is not intended for that and many properties are not reinitialized to defaults.
- *PL_Save* (areaRef:L; XML:T) -> result:L
 - save the area settings into XML
 - note: Use *PL_Save* after configuring the area but before printing. PLP modifies properties during printing - you would not get the original settings.

PrintList Pro version 5 – New Break Processing commands: Computed Breaks

- *PL_ProcessArrays* (callbackMethodName:S; breakArrays:Y; dataArrays:Y; useDetail:L) -> error:L
 - both breakArrays and dataArrays have to be ARRAY POINTER with pointers to arrays (can use local arrays).
 - first element in breakArrays determines break level 1, second 2, ...
 - first element in dataArrays is column 1 for *PL_GetBreakValue*, second is 2, ...
 - set useDetail to 0 to be called only with break information, set it to 1 to be called with every data row and with breaks.
- *PL_GetBreakValue* (handle:L; column:L; calculation:L) -> value:F
 - handle is an identification of the current *PL_ProcessArrays* call (currently process ID is used -> don't call *PL_ProcessArrays* in the callback!)
 - column is the index into the dataArrays
 - calculation can be one of the following values:
 - 1 - Sum
 - 2 - Minimum
 - 3 - Average
 - 4 - Maximum
 - 5 - Count
 - 6 - Variance
 - 7 - Standard deviation
 - when column is out of range, zero is returned.
 - callbackMethodName is called as: *callbackMethodName* (handle:L; row:L; breakLevel:L)
 - handle is to be used as parameter 1 in call to *PL_GetBreakValue*
 - row is the current row number
 - breakLevel is the current break level, -1 for data row (if useDetail = 1)
 - only in the callback you can call *PL_GetBreakValue*, use the provided handle.
 - *PL_ProcessArrays* operates on a copy of the arrays, you can freely modify them in the callback.

- when breakArrays is empty, the only break generated will be 0.
- when dataArrays is empty, no calculations are performed.
- calculations are performed "on the fly" - after fetching each data row. So when you call *PL_GetBreakValue* on some data row (not on a break), you will get current values for that row. E.g. using empty breakArrays and useDetail=1, calling *PL_GetBreakValue* to get SUM, you get sum of rows 1 through current row.

PrintList Pro version 5 – New break calculations

In break calculations, "\Var" will compute variance and "\Dev" will compute standard deviation.

PrintList Pro version 5 – Calculated columns in array mode

PrintList Pro now supports calculated columns in array mode, when previous versions only offered this feature in field mode.

To make a column calculated, create a regular array-based column and then use:

PL_SetCalcCall (\$area; \$column; \$methodName)

- The callback is expected to be declared as Handler (area:long; column:long; type:long; ptr:pointer; first:long; count:long)
- The callback has the same parameters as a column callback in fields mode, but the array is fully sized (by you as developer), you have to fill the requested elements; the type is the actual array type, not a field type (e.g. 'Integer Array' instead of 'Is Integer')

PrintList Pro version 5 – New RGB commands

- *PL_SetBrkColRGBOpt* (areaRef:L; breakLevel:I; columnNum:I; showColDivider:I; lineWidth:F; dividerRed:L; dividerGreen:L; dividerBlue:L)
- *PL_SetBkHColRGBOpt* (areaRef:L; breakLevel:I; columnNum:I; showColDivider:I; lineWidth:F; dividerRed:L; dividerGreen:L; dividerBlue:L)
- *PL_SetBrkRowRGBDiv* (areaRef:L; lineWidth:F; dividerRed:L; dividerGreen:L; dividerBlue:L)

PrintList Pro version 5 – Additions to existing commands

- Implemented support for picture alignment when format is 0 or 3 in *PL_SetFormat*.
 - picture formats:
 - "0" (or not specified or out of range) is "Truncated (non-centered)"
 - "1" is "Truncated (centered)"
 - "2" is "Scaled to fit"
 - "3" is "Scaled to fit (proportional)"

- "4" is "Scaled to fit centered (prop.)"
 - formats 1 and 4 are always centered, format 2 fills the whole rectangle.
 - thus only formats 0 and 3 are handled specially - both horizontal and vertical alignments are honored.
 - default alignments are centered.
 - Note: when variable row height is used, Picture columns are used for row height calculation, too (even when usePictHeight in *PL_SetFormat* was set to zero)
- *PL_SetColOpts* has a new parameter at the end:
 - drawing engine (integer): 0 = use GDI plus for drawing (default), anything else: use GDI for drawing (ignored on Mac)
 - > *PL_SetColOpts* (areaRef:L; hideLastColumns:I; hideDetailArea:I; drawingEngine:I)
- *PL_SetFormat* has three new parameters at the end:
 - attributed (integer): 0 = plain text (default), anything else = attributed (multi-style) text
 - note: the multi-style property is applied to all formatting in that column -> breaks have to account for that!
 - lineSpacing (real): line spacing to use, 0 is substituted by 1.0 (default)
 - vertical alignment (integer): 0 - default, 1 - top, 2 - center, 3 - bottom
 - > *PL_SetFormat* (areaRef:L; columnNumber:I; format:S; columnJust:I; headerJust:I; usePictureHeight:I; attributed:I; lineSpacing:F; vertAlignment:I)
- Double lines (typical in accounting) are now supported - just use 2.0 as the lineWidth (5th parameter in call to *PL_SetBrkColOpt* / *PL_SetBrkColRGBOpt* / *PL_SetBkHColOpt* / *SetBkHColRGBOpt*): two 0.25 pt lines will be drawn.
- *PL_Register* can accept a new optional parameter
 - *PL_Register* (key:T; forceCheck:L) -> licenseError:L
 - if forceCheck is nonzero, the provided key is tested regardless of current registration state. If the plugin was not previously registered and the result is 0, it is registered the same way as if forceCheck was zero (or omitted).

PrintList Pro version 5 – Printing an AreaList Pro area

To print AreaList Pro areas, just call *AL_Save* and *PL_Load* – they use the same XML UTF-8 format. Then set row/cell options and you are done. Or add break processing options.

Using this feature will allow you to indirectly apply AreaList Pro v9 numerous formatting options (using the v9 property based API) to a PrintList Pro area using an AreaList Pro area as the data and formatting source. Note that this feature is only available for persistent AreaList properties. Refer to AreaList Pro manual for property persistency.

Upgrading from previous versions

PrintList Pro version 5.3 is a free upgrade from v4.7 and above: existing v4.7/v5.x licenses will activate version 5.3.

Upgrades from version 4.6 (single user or server) are available at 50 € (\$50-\$55 USD).

Contact

Upgrade requests, sales and license information: orders@e-node.net

Technical issues and requests: <http://forums.e-node.net>

PrintList Pro web page: <http://www.e-node.net/plp>