

AreaList Pro

Version 11.3 Release notes

April 24, 2023

Published by e-Node worldwide
www.e-node.net/alp

System requirements

AreaList Pro version 11 is compatible with 4D v18 and over for MacOS (10.14 or over, Intel and Silicon, signed and notarized) and Windows (64-bit mode).

This version also supports v8 API and v9-10 API: your existing code should not require any changes.

Version 11.3 – New

- • Added a new property for scrollbar behavior
 - name ALP Area ScrollbarStyle
 - value "scst"
 - type Boolean
 - read/write/persistent
 - xml name "scrollbarStyle"
 - default 0
 - -> 1 means "use overlay mode" - visible only while scrolling (with mouse wheel), scrollbar does not consume space in the area, it is drawn temporarily over the data
- Added persistency for property ALP Area ScrollbarArrowsPos, xml name "scrollbarArrows"
- Added property for limiting long text
 - name ALP Area MaxRowLines
 - value "rowm"
 - type integer
 - read/write/persistent
 - xml name "maxRowLines"
 - default 0, min 0, max 64

When ALP Area NumRowLines = 0 (variable row height) and this property is set to a value greater than 0, at most ALP Area MaxRowLines will be displayed

In that case when ellipsis is to be used (e.g. ALP Column UseEllipsis), last line will display it.

NOTE that ellipsis addition to last line is implemented only for left aligned text (default alignment for text is mapped to left), does not work for any other alignment (where ellipsis is at the beginning or in the middle of a long line)

ALP Area MaxRowLines differs from ALP Area MaxCellHeight in:

- number of lines of text instead of points
- ellipsis for left aligned text can be drawn at end of last visible line

Example:

Having [left aligned text] "111\r222\r333" and ellipsis active (e.g. ALP Column UseEllipsis)

ALP Area NumRowLines = 1 (old AreaList Pro v7 behavior: ignore \r, draw single line)

111222333

ALP Area NumRowLines = 0, ALP Area MaxRowLines = 0:

111

222

333

ALP Area NumRowLines = 0, ALP Area MaxRowLines = 1:

111...

ALP Area NumRowLines = 0, ALP Area MaxRowLines = 2:

111

222...

ALP Area NumRowLines = 0, ALP Area MaxRowLines >= 3:

111

222

333

- Added property for getting current system highlight color
 - name ALP Area HighlightColor
 - value "uihc"
 - type integer
 - read/write
- ALP Area HighlightColor is writable, this color will be used in place of the system highlight color
 - to reset it to use the system color again, set the value to 0
 - Note: this is a global property, it will affect all areas
- Using native appearance (ALP Area Appearance = -1) on Mac finally supports accent color for checkbox/radio button
- Added a new command to get a picture as used internally in AreaList Pro
 - *AL_GetIcon* (areaRef:L; iconID:L) -> Picture
 - current appearance of the area is used to resolve special icon IDs (areaRef can be zero)
 - can be used e.g. to get native appearance pictures (ALP Area Appearance = -1) and use them with pictures appearance (ALP Area Appearance = 11)

- Added constants for miscellaneous internal icons (popups, node disclosure triangle, checkbox, radio button, sort indicator)
 - AL Icon Generic Popup PictID
 - AL Icon Date PictID
 - AL Icon Time PictID
 - AL Icon Node Right PictID
 - AL Icon Node Down PictID
 - AL Icon CB Unchecked PictID
 - AL Icon CB Checked PictID
 - AL Icon CB Mixed PictID
 - AL Icon RB Unchecked PictID
 - AL Icon RB Checked PictID
 - AL Icon Sort Up PictID
 - AL Icon Sort Down PictID
 - AL Icon Appearance Offset

When using *AL_GetIcon*, current ALP Area Appearance is used to resolve the real picture ID, but when setting the icon, nothing special is done, you must use AL Icon Appearance Offset to modify icons for ALP Area Appearance = 11 (AL Use Pictures Interface).

Version 11.3 – Changes

- Added support for Win11 native icons (header, checkbox, radio button, scrollbars)
- Added new appearance constant for Win11 - name AL Force Win11 Interface, value 12
- Changed maximum allowed value for ALP Area Appearance from 11 to 12
- Header drawing changes:
 - fixed header drawing
 - -> no divider separator above the vertical scrollbar
 - added possibility to not draw the divider separator for last column (on the right)
 - name ALP Area HideLastHdrDivider, value "hhld", type Boolean, read/write/persistent, xml name "hideLastHdrDiv", default 0 (False)
- Changed drawing of headers, especially on Windows (try using ALP Area WindowsText > 0)
- Changed position of popup icon computation (it is not always at top, depends on line height and vertical alignment)

• Version 11.3 – Fixes

- Fixed a possible crash in old API (e.g. using AL IsValidArea with area 0)
- Fixed slow drawing ("row by row") after hierarchy is sorted in Fields mode

- Fixed handling of arrows in scrollbar: when using **GOTO OBJECT** in a form and then displaying a new form with AreaList Pro, it receives dozens of IDLE events - too fast scrolling
- Fixed entry mode on Windows when Windows scaling is used (with 4D v19 Rx; Windows scaling support implemented in 19R4)
- Fixed possible freeze parsing malformed styled text
- Fixed endRow for "\\FUNCTION" callback in break header when "\\function" is the only "break value" used in the break header cells
- Fixed printing of repeating break values when configured to be omitted (not drawn)
 - -> same behavior as old PrintList Pro
- Fixed column sizing when ALP Area ResizeDuring is in effect and data are updated
- Fixed bad header drawing after a column resize as effect of a double-click on header column divider

Version 11.2.2

- Fixed printing of breaks

Version 11.2.1

- Fixed a crash when invisible rows are used and user clicks into empty area below data (bug new to ALP 11.2)
- Fixed handling of invisible columns (would fail for EntitySelection) for D&D and Copy

Version 11.2 – New

- Added missing constant and accessor for default time format (which defaults to "0")
 - name ALP Area DefFmtTime, value "dfti"
 - when adding a column using time field or array time, this format is used as default (added for 4D v14 ARRAY TIME back in 2013)
- Added ability to export only defined columns:
 - set the list of columns to export (using ALP Area ExportList or ALP Object ExportList)
 - when setting ALP Area ExportOptions, include the AL Export Using List option to use the list of columns
- Added constants for that:
 - added area property ALP Area ExportList, value "expL", type text, read/write/persistent, xml name "exportList", default ""
 - use comma separated list of column numbers, e.g. "11,3,4,9,1" -> export only these columns in this order

- **Note:** when moving columns (by D&D (and ALP Area Compatibility = 1 or ALP Area DragMoveColumns = 1) or with AL_MoveColumn), this list is automatically adjusted (column numbers are renumbered according to the move)
 - added constant ALP Object ExportList (value "expL") to be used with AL_GetObjects/AL_SetObjects
 - expected is ARRAY INTEGER or ARRAY LONGINT containing the column numbers
 - added constant AL Export Using List (value 65536) to be used in ALP Area ExportOptions property
 - when this value is used and ALP Area ExportList is not empty, only columns defined in ALP Area ExportList will be exported with ALP Area ExportToFile
 - changed maximum value allowed for ALP Area ExportOptions: max = 131071 (0x0001FFFF)
- Added new property to allow sorting selection by calculated column in Fields mode
 - name ALP Column SortFormula
 - svalue "cobf"
 - type text
 - read/write/persistent
 - XML name "sortFormula"
 - default ""
 - Set this property to a method name which will be called by ORDER BY FORMULA as *Method* (\$alp:int; \$colNum:int) -> value to use for order
 - Note that \$colNum is negative for descending order - ORDER BY FORMULA explicitly uses descending order, you should not revert it
 - To be usable by end user (click on header or Sort Editor), you must explicitly set ALP Column Sortable to 1 (calculated columns are not sortable by default).

e.g.

```

$err:=AL_AddCalculatedColumn($area; $colType; "MyColumnCallback")
AL_SetColumnTextProperty($area; $col; ALP_Column_SortFormula;
"MyColumnSortCallback")
AL_SetColumnLongProperty($area; $col; ALP_Column_Sortable; 1)

// MyColumnSortCallback
C_LONGINT($1) // ALP
C_LONGINT($2) // column number, negative when descending order
will be used (using explicit ">"; "<")
// C_xxxxx($0)
$0:=MyCalculatedValueForCurrentRow.call($1; Abs ($2))

```

Version 11.2 – Changes

- Added support for hidden columns used in hierarchy
 - first visible [data, not the one defining the first hierarchy level] column properly displays data when it ends as first column even when using auto-hierarchy
- Added support for auto-hierarchy without creating the break headers
 - when first column in the list set with ALP Area AutoHierarchy/ALP Object AutoHierarchy is negative, AreaList Pro does not change the break headers, you can use custom headers (e.g. with \SUM in any data column)
- Format is now resolved recursively (up to 5 times)
- Export (Excel) does not use format when it is not resolved (starts with '|' or ':')
- Added auto-width computation for break rows, including total displayed in footer row
- Relaxed *AL_DuplicateColumn* requirements:
 - old and new data pointer can use different data type
 - developer is responsible for properly re-setting relevant properties (e.g. ALP Column Format)
- Added implementation for formats when using 4D in project mode (by parsing the main DB's filters.json file)
 - formats should be properly resolved (needed for Excel export)
- Changed Excel export to set Wrap mode implicitly when ALP Area NumRowLines is not 1
 - Excel will not wrap when text contains "\r" unless wrap mode is enabled

Version 11.2 – Fixes

- Fixed "syntax error: collection or object expected" alert when retrieving field type from "This.relatedEntity.anotherRelatedEntity.entityName"
- Fixed various issues related to breaks
- Fixed drawing of column separators in break rows
- Fixed selection with up/down arrow keys (esp. when break processing with hierarchy is used)
- Fixed possible crash when changing selection (and clearing data cache) while break processing is used
- Fixed auto-width computation for hierarchy when [leading] break columns are invisible
- Fixed multiple-row selection mode (anchor row not properly set with drag-select, next keyboard navigation (using shift key) selects incorrect row)

- Fixed an old bug related to 4D v11+ when "use comma and period as placeholders" is in effect
 - when it is active, default column format must not be using localized version of the format, but the US format (e.g. even for French locale use "#,###,##0.00", not "# ### ##0,00")
 - -> non-localized format is used as default column format for integer, longint and real (set at startup)
- Fixed crash (memory corruption) when using *AL_SetIcon* to replace existing picture (bug from ALP 11.0b10)
- Fixed binary Advanced Properties loading in Design (didn't include any column) (bug new in ALP 11.0b11)
- The area properly redraws when setting hierarchy expansion state programmatically
- Fixed implementation for ALP Area PrintArea
 - it was failing to properly initialize the printing area when arrays mode was used with arrays in different scope (ALP in main DB but pointers to arrays from a component or vice versa)
- Fixed drawing when many rows are invisible in Fields mode (by not fetching invisible rows from records)
similar (identical, but in different part) problem was fixed in 10.3b2 - ALP in Arrays mode
- Fixed drawing of headers
- Fixed bad scrolling during entry (old ALP 10.1 bug, at specific conditions, an area could scroll the entry cell out of view on every key down)

Version 11.2 – Windows specific

- Modified property ALP Area WindowsText from type Boolean to type integer, min = 0, max = 2, default = 0
 - 0 - GDIPlus
 - 1 - GDI
 - 2 - DirectDraw + DirectWrite
- Changed handling of pictures (resolution/DPI, new implementation for conversion for GDI & DD)
 - -> could be the icons/pictures are too small (there was no conversion for 72 dpi vs. 96 dpi, now there is one)
- Fixed Windows scaling (4D v19R4+)
 - -> click into area, entry widget position, text size, pictures, ...
 - scrollbars and headers are drawn differently to get rid of white hair lines between elements

Version 11.1

- Fixed type-ahead
- Fixed ALP Area Name getter
- Fixed header "disappearing"
 - under some specific conditions, cells in header were not drawn
- Setter for ALP Area TypeAheadString now executes the type-ahead in focused target when possible - similarly to user typing the specified text
 - now, you can scroll/select row in AreaList Pro with code, e.g. assuming Example 15 is sorted by city, you can select a row:
`AL_SetAreaLongProperty(eList; ALP Area SortList; 1)`
`AL_SetAreaTextProperty(eList; ALP Area TypeAheadString; "Orlando")`
- Added properties to ALP Area PrintOptions object for standalone printing
 - "drawFrame": int, 0-2 — when present and in range, ALP Area DrawFrame is set to this value, otherwise ALP Area DrawFrame is set to 1 when it is 2
 - "printHeaders": int, 0-2 — when present and in range, PLP PrintHeaders is set to this value, otherwise PLP PrintHeaders is set to 0 for ALP Area HideHeaders=1 and 2 for ALP Area HideHeaders=0
 - "frameWidth": real, 0.0-1.0 — when present and in range, PLP FrameWidth is set to this value (nothing is done otherwise)
- PLP FrameWidth default is 1.0 and is not modified when printing
 - to be used, ALP Area DrawFrame must be 1
- Changed maximum allowed value for ALP Column Enterable and ALP Cell Enterable, maximum is now 6
 - When cell enterability is 6, click anywhere into the cell is handled the same as click on the popup icon
- Fixed popup icon click detection - possibly wide column separator is now honored

Version 11 – Major new Features

- **Sub-ALP** is the most fantastic new feature, that you won't find anywhere else. A column displays cells that really are AreaList Pro areas with their own formatting, like invoices for a customer, invoice lines for an invoice, etc.
- **Collections** are now supported for data source.
- **Entity Selections**: now you can integrate 4D's ORDA functionality into AreaList Pro! Use an Entity Selection in your AreaList Pro area.
- **PrintList Pro** is now included as a license option: only one plugin instead of two (no change of PrintList Pro code needed).
- **Standalone Printing**: in addition to PrintList Pro or SuperReport Pro printing of an AreaList Pro area, a single new command just prints the area with all formatting.

- **Export to Excel** (XLS or XLSX) including formatting, with or without header, selected rows or whole area, and many more options (including Sub-ALP).
- **Offscreen AreaList Pro** areas: build an area and Print it or export to Excel without displaying it on screen, with a single command.
- **Apple Silicon** (M1, M2) support.
- By popular demand, the return of the **PDF Manual**, fully updated, including PrintList Pro and detailed explanations of v11 new features.

See the [manual](#) and the [forum](#) for additional details and Hints & tips.

Upgrading from previous versions

Any v11 license key will activate any v11.x version.

A specific Print license option is available to enable printing with the PrintList Pro module or Standalone. Old PrintList Pro plugin must be removed from the Plugins folder.

Upgrades from versions 9.x and 10.x (single user or server) are available at www.e-node.net/alp.

License extensions to SuperReport Pro are also available (all types, including server licenses).

Contact

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

Technical issues, discussions and requests: forums.e-node.net

AreaList Pro web page: www.e-node.net/alp