

AreaList Pro

Version 10.3.1

Release notes

March 19, 2020

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

System requirements

AreaList Pro version 10 is compatible with 4D v16, 4D v17, v18 and above, for macOS and Windows, in both 32-bit and 64-bit modes. It is reported, but not certified, to run with 4D v15 in 32-bit mode.

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

A specific Signed & notarized "ALP64" version is available for 64-bit only setups (requires 4D v17.4 or v18+).

Version 10.3.1

- Fixed ALP Area SelNone=1 handling when ALP Area SelMultiple=0 (was not visually deselecting the row on cmd-click)
- Row computation/reporting when using ALP Area DontSortArrays (implicit on hierarchy sorting), includes entry cell computation and row revealing
- Fixed visual row computation when using breaks
- New template for default printing an AreaList Pro area with SuperReport Pro includes page count (and total)
- Fixed entry of F1-F12 and dead key - ignored to not insert non-text
- Signed & notarized 64-bit version (requires 4D v17.4 or v18+)

Version 10.3

- 64-bit only specific bundle available
- Fixed row selection when drag & drop is enabled, and multiple rows are selected: clicking on an already selected row did nothing
 - that row now should be the only selected row
- Fixed a bug with many hidden rows and big selection - should not "freeze" on every click/redraw

- When ALP Area AutoResizeColumn (or old ALP Area AutoSnapLastColumn) is in effect, the column's minimum width (ALP Column MinWidth) is honored
- Fixed scrolling and cell selection when ALP Area SelType > 0 (cell mode), ALP Area ColsLocked > 0 and a click occurs in the locked columns section
 - When tracking starts in locked columns:
 - if you stay inside locked columns, it won't scroll horizontally
 - if you move to unlocked part, the area scrolls to the first unlocked column (otherwise x-th column would be selected, not the first right to the locked area)
 - When tracking starts in unlocked area, it scrolls as expected
- Fixed entry mode when display mode does not match entry mode (e.g. checkbox vs. radio buttons), it should draw only entry-relevant format (and ignore first click)
- Auto-hierarchy related fixes:
 - incorrect row was used for the entry, e.g. double-clicking second row (first data row) started editing in third row
 - fixed sorting of hierarchy in arrays mode
 - first column draws everything except the actual value (was not drawing anything from cell options)
- Fixed keyboard handling in 64-bit on Mac - keys in "Private Use Area" (0xE000-0xF8FF) are ignored (e.g. for type-ahead)
 - → F1 works the same as in 32-bit (where the F1 key does not report valid Unicode character)
- Fixed focus drawing in entry mode for checkbox/radio buttons/picture
 - → focus was drawn only after text entry or window deactivation/reactivation
- Fixed checkbox/radio buttons entry when the respective picture is bigger (in height) than actual row height
 - → picture was drawn clipped, label was moved down; now the picture is drawn scaled, label is at top

Version 10.2.1

- Fixed cell-selection mode mouse tracking (bug introduced in 10.0)

Version 10.2 – New features

- New property for checkbox focusability (when displaying checkboxes)
 - name ALP Column FocusableCheckbox
 - category Column
 - value "eccf"
 - type Boolean
 - default 0

- XML name "checkboxFocusable"
- read/write/persistent
- when set to 1, a column displaying checkboxes or pictures is focusable as any other column type
- Modified *AL_SuperReport* to set SRP Column ExecutionLevel to 1 for related fields
 - SuperReport Pro 4.2 can print related fields referenced in the Table object
 - Note: SuperReport Pro Tables do not (yet) support AreaList Pro feature of displaying only related fields from a main table (no field for main table included in the view)
- Added possibility for DisplayList to not select the first row when multiple row selection is used (first parameter to *SetListBehavior* is 1)
 - Previously, when provided selection array was not selecting any row, first row was selected, now if the array is not empty, the first row is not selected
 - → append an invalid value to the empty selection array (e.g. 0) used in *SetListSelect* to have no rows selected when displaying the list
- New event and constant for key down: AL Key event (value 19, callback only)
 - For the key pressed, look for ALP Area EventKey, ALP Area EventChar, ALP Area EventModifiers

Version 10.2 – Changes

- Removed unnecessary redraws during drag
- Changed area scrolling during drag (was too slow)
- Improved bidirectional text support
- XML for area now includes row and cell options
 - → usable for printing AreaList Pro data with PrintList Pro

Version 10.2 – Fixes

- Fixed event reporting for Drag & Drop
 - Drag & Drop inside an AreaList Pro area will very likely report Drop, then Drag; event order for drag from one AreaList Pro area and drop to another one (especially across processes) is not defined (depends on 4D execution)
 - this is true for object method execution, if using event callback, the call is made when the event is to be reported, not when 4D allows to execute the object method
- Fixed drop handler - when a column from a group (ALP Column GroupID) is dragged outside the group, drag handler properly displays it is not allowed (no column is highlighted), but drop (if the user released the mouse there) was handled as drop in front of all columns - now the drop is simply ignored

- Fixed Wrap property in Entry mode - in compatibility mode (ALP Area Compatibility = 1), the wrap is set by AreaList Pro depending on ALP Area NumRowLines, but it was not honored during entry (you got single-line even when area was set to display e.g. 6 lines)
- Fixed a bad break row height computation when the number of row lines is fixed (ALP Break NumRowLines > 0) and zoom is in effect (ALP Area Zoom # 1.0)

Version 10.1 – Introducing the native entry mode

We have reduced AreaList Pro's dependency on 4D for cell entry mode.

Version 10.1 uses a native technology to edit enterable cells, with the result of completely rewriting all the cell editing user interface.

This implementation removes the side-effects of using a small 4D popup type window as in v10.0, although this mode is still available with ALP Area EntryMethod.

Version 10.1 – New features

- Highly improved redrawing speed
- New ALP Area EntryMethod: use the 10.0 entry widget according to data type
 - value "enem", read/write, type integer, default 1 or 7 (depends on current locale), min 0, max 15
 - bit 0 (value 1) = styled text
 - bit 1 (value 2) = plain text
 - bit 2 (value 4) - integer/real/date/time
 - bit 3 (value 8) - picture
 - -> default for LTR system is 1, default for RTL system is 7
 - checkbox, radio buttons and "draw integers as pictures" are always handled directly, not in the 10.0 entry widget
 - not using the 10.0 entry widget for styled text (i.e. bit 0 false) means the 10.1 native editor is used and the user will edit "text", not styled text.
 - using the 10.0 entry widget means you can use ALP Column EntryFormat, but you can't have keyboard shortcuts executed for your form during editing
- Here is how enterable cells will behave according to ALP Area EntryFirstClickMode with ALP Area EntryMethod default value:
 - ALP Area EntryFirstClickMode = 0, cell value does not matter: that click is interpreted by the widget -> no selection, caret is somewhere in the text
 - ALP Area EntryFirstClickMode = 1, value is "not empty" (e.g. 123.45) - identical to ALP Area EntryClick = 0
 - ALP Area EntryFirstClickMode = 1, value is "empty" (e.g. 0.00) - that zero is selected, caret is at the end
 - ALP Area EntryFirstClickMode = 2, value is "not empty" (e.g. 123.45) - that value is selected, caret is at the end

- ALP Area EntryFirstClickMode = 2, value is "empty" (e.g. 0.00) - that zero is not selected, caret is where you clicked
- ALP Area EntryFirstClickMode = 3, cell value does not matter: that click is not interpreted by the widget -> cell value is selected, caret is at the end
- Copy enabled in arrays mode when there is a currently selected cell (in single cell selection mode)
- Copy (and thus D&D) modified to include all data for selected rows/cells even in field mode (this can take a while for large selections!)
- Implemented Copy for multiple cell selection mode (was copying only the selected cell)
- Implemented Copy action in field mode: when an area is in field mode, Copy is always disabled – if the developer overrides that, executing Copy did nothing, now it copies the selection (same as when performing D&D)
- Maximum column width can be set with ALP Column MaxWidth:
 - value "widM"
 - xml name "maxWidth"
 - read/write/persistent
 - type real
 - default 32000
 - min 0
 - max 32000
 - when smaller than ALP Column MinWidth, ALP Column MinWidth is used
- New property: "proportional column resizing" can be set with ALP Column Proportional
 - to be used along with ALP Area AutoresizeAllColumns (see below)
 - scope Columns
 - value "cwid"
 - xml name "relativeWidth"
 - read/write/persistent
 - type Boolean
 - default 0 (False)
- New property: ALP Area AutoresizeAllColumns
 - set the ALP Column MinWidth, otherwise the columns will be 0 points wide!
 - value "Snap"
 - xml name "autoResizeAllColumns"
 - read/write/persistent
 - type integer
 - default 0
 - min 0
 - max 2
 - 0 = none

- 1 = LB like (shrink to fit)
- 2 = HTML like (proportional resizing of marked columns):
- to have any effect, at least one visible column must have ALP Column Proportional set to 1!
- only visible columns with ALP Column Proportional property set to 1 are resized
- ALP Column Proportional means that the difference (shrink/expand) is distributed proportionally across the columns (if possible - honoring min/max width) using current column width
- e.g. if you have one column with width 100, another one with width 200, the first one will get 1/3 of the resize, the second one will get 2/3 of the resize
- if only one visible column has this property set, it behaves similarly to ALP Area AutoResizeColumn
- Added support for enterability of related fields
 - record from main table is loaded with automatic relations set to true
 - if both records are unlocked, entry is allowed
 - entry is edited as usual
 - related record is saved
 - record from main table is saved
- New property to set the color of the empty area below the last row:
 - name ALP Area EmptyRowColor
 - category Display
 - value "emrc"
 - type color
 - default 0
 - when set to non-zero, used to color empty rows below last data row

Reminder:

- to hide the column dividers in this area: set ALP Area ShowColDividers bit 3 to 1 (add value 8)
- to hide the row dividers in this area: set ALP Area ShowRowDividers bit 1 to 1 (add value 2)
- Added support for RTL/bidirectional text
- Added new event for horizontal scroll:
 - AL Horizontal Scroll Event
 - type integer
 - value 16
- Modified ALP Area Event Filter: added 3 new bits, maximum is now 15
 - bit 1 (value 2): don't report keyDown/autoKey (if bit 1 is zero, keyDown event is reported to the callback)
 - bit 2 (value 4): don't report vertical scroll

- bit 3 (value 8): report horizontal scroll
- Pending event is now an array, not a single event
 - -> ALP Area SendEvent returns the first event in the queue (getter), adds requested event to the end of the queue (setter)
 - events are sequentially reported
 - **Note:** when an event callback (ALP Area CallbackMethOnEvent) is installed, the event is reported immediately (except during update), the object method may fail to be executed... use event callbacks!
 - horizontal and vertical scroll events are cumulated and moved to the end of the queue (e.g. using mouse wheel, many scroll events can be reported, but actually the object method can't be executed during mouse wheel event and the event is reported at next idle -> will be reported only once, not many times after ending the scroll)
 - two identical events are never put into the queue

Version 10.1 – Changes

- ALP Area ResizeDuring honored – no column resizing when scrolling/entering data unless ALP Area ResizeDuring is set to 1
- Faster column width/row height computation
- Better column width computation for fixed-size columns when ALP Area Zoom is not 1.0
- Changed default and minimum column width (ALP Column MinWidth) from 5 to 0
- Resizing a column with mouse honors minimum and maximum column width
- Dot on numeric keypad really produces comma/dot (depending on regional settings for decimal point) instead of keyboard definition
- Note: we can make it conditional and available even on macOS if it makes sense (0 - use keyboard (default for Mac), 1 - use regional settings (default for Windows), 2 - use regional settings, but only for real number input, not for date/time/text entry) – Feel free to ask!*
- Changed minimum for ALP Area DragScroll: can be set to -1, meaning "no scroll" (both directions) when dragging row/cell
- Slightly changed scrolling speed during D&D: should scroll slower
- Improved speed when using big text drawn in a single line (when not using ellipsis, only the visible part is measured, not whole text)
- When a text has to be drawn in single line, Carriage Return is not interpreted (does not start a new line), but simply printed (output depends on the font used, usually a space) - as in the old AreaList Pro v8 behavior
- Optimized drawing (not requesting a redraw when nothing changed - e.g. click -> select row -> redraw, second click on the same row -> no redraw)
- Implemented workaround for a new 4D v16+ specific bug
 - double click should be reported

- use `AreaList Pro` event callback instead of object method - this works
- Changed scrollbars
 - there are scrollbars for Mac, XP, Win7, Win10, not only for Mac and Win10 (Mac has only one appearance - OSX 10.7+)
 - the pictures are created for default Blue appearance
 - scrollbars use 15 (11 for small scrollbars) points for Mac, 17 (13) for Win
 - when system default is used (`ALP Area Appearance = -1`) and system supports it (e.g. not using Classic appearance on Win), current native scrollbars are used (e.g. Olive coloring on Win10), the actual scrollbar size as defined by the system is used for normal scrollbars (13 for small scrollbars on Win is used)
- Formatting workaround again - using `PA_String` instead of `PA_FormatXXX` - to match 4D's String (which differs at least for Date with format Internal date short special)
- Slightly modified drag & drop, especially from `AreaList Pro` to `AreaList Pro` across windows (processes)
- Changed ALP Column Sortable
 - default is always 1: is not cleared for non-sortable data (e.g. pictures) on column creation
 - otherwise it is a change in behavior: getting header click instead of user sort when using `ALP Area UserSort = AL User sort bypass`
- Changed creation of native pictures on Windows
 - when using native appearance (`ALP Area Appearance = -1 / default`), checkboxes, headers, scrollbars, ... are created using native drawing
 - `AreaList Pro` tries many times to get the picture and if it fails, default for the platform is used (same as using `ALP Area Appearance = 0`)
- Implemented workaround for inability to call object method on mouseUp (4D bug ACI0098952)
- If a font on OSX does not support italic, it is emulated using font transformation
- Styled-text entry now uses implementation from `AreaList Pro v10.0`: popup window, not modal dialog (from `AreaList Pro v9`)
- Removed default filters during entry (inability to input text like "today" into date cell)

Version 10.1 – Fixes

- ALP Area DragProcessID now really returns the ID of the process initiating the D&D, not of the area process
- Fixed `DisplayList` (ignoring buttons/shortcuts, dialog positioning, ... - new bug in `AreaList Pro v10.0` related to needed recoding for 4D v14.4 internal components)
- Fixed getters for Drop destination area when the destination is `AL_DropArea` (`ALP Area DragDstArea`, `AL_GetDrgArea`, `AL_GetDragLine`, `AL_GetDragCol`)

- Fixed creation of pictures when using native appearance (ALP Area Appearance = -1) on Windows 10

Version 10.0.1 – New features

- Added property to ignore mouse wheel
 - ALP Area IgnoreMouseWheel, value "eimw", type int, read/write, default 0, min 0, max 3
 - used as bitfield:
 - bit 0 (value 1) - ignore horizontal scrolling with mouse wheel
 - bit 1 (value 2) - ignore vertical scrolling with mouse wheel
- Added property for entry focus hiding
 - ALP Area HideEntryFocus, category Entry, value "Efoc", type int, default 0, min 0, max 3
 - used as bitfield:
 - bit 0 (value 1) - don't draw focus rectangle for text entry (text, int, real, date, time)
 - bit 1 (value 2) - don't draw focus rectangle for other entry types (checkbox, radio, picture)

Version 10.0.1 – New constants

- Added constants related to some new v10 features (break processing, alternate header, transposition):
 - ALP Area ClickedVisualRow ("evcR") - when using break processing, the second visual row is very likely the first data row (break header is first row)
 - ALP Area ClickedRowNum ("evCr") - the real row number (differs from ALP Area ClickedRow only when using ALP Area Transposed)
 - ALP Area ClickedColNum ("evCc") - the real column number (differs from ALP Area ClickedCol only when using ALP Area Transposed)
 - ALP Area RollOverVisualRow ("evlR") - similar to ALP Area ClickedVisualRow, but for Mouse Over
 - ALP Area RollOverRowNum ("evLr")
 - ALP Area RollOverColNum ("evLc")
 - ALP Area EntryCellRow ("ecCR") - visual cell row number (can differ when using break processing or transposition)
 - ALP Area EntryCellColumn ("ecCC") - visual cell column number (can differ when using transposition)
 - ALP Area AltHdrRowsInGrid ("rowA") - number of rows in alternate header grid
 - ALP Area AltHdrClear ("AlHC") - clear the alternate header grid & cells

- ALP Object AltHdrGrid (“GRIA”) - alternate header grid (see grid manipulation) usable with *AL_Get/SetObjectsX*
- Added new selectors/constants for direct access to break cells (in ALP Break Properties theme):
 - ALP BreakCell Source (“src”) - data source for the cell (default is inherited from the data grid)
 - ALP BreakCell Value (“valu”) - alias to ALP Cell Value
 - ALP BreakCell ColSpan (“gcsp”) - colSpan of the break cell grid
 - ALP BreakCell RowSpan (“grsp”) - rowSpan of the break cell grid
- Added missing constants:
 - ALP Break LineShowColumnDivider (“PLsd”)
 - ALP Break LineThickness (“PLlt”)
 - ALP Break LineColor (“PLlc”)
 - ALP Break Clear (“clr”)

Version 10.0.1 – Changes

- Changed column lock divider drawing to be only one point wide
- SQL columns use column name, not original column name (using the alias set by "AS" in the query)
- Changed behavior for breaks
 - when accessing cell lines, cell number is expected, not column number
 - all other places (cell formatting) expect cell numbers instead of column numbers when used with alternate header or break rows
- When setting break property, area reference can be -2 or -3 (all AreaList Pro areas, all AreaList Pro areas in current process) and break level can be -2 (all existing break levels)
 - example: to clear all break footers
AL_SetBreakLongProperty (\$area;0;-2;0;ALP Break Clear)
- Column separators are drawn after a row is drawn, not after the whole area is drawn
 - break row column separators should work as expected
 - added new bit to ALP Area ShowColDividers: setting bit 3 (value 8) to 1 will not draw the column dividers in empty area (below data rows)
 - changed maximum to 15
- Added missing PrintList Pro-inspired break properties into XML generation
- Modified *AL_ModifyArrays* to be usable in fields mode (for row/cell options move)
 - ➔ See the [documentation](#) for a detailed explanation.
 - ➔ Clicking on a calculated column header no longer displays the 4D sort editor
- Tooltip is created when text or cell changes, not on every mouse move (should not flicker)

- Added support for ALP Column CallbackFnc in break processing (only ALP Column Callback was supported for break data/values)

Version 10.0.1 – Fixes

- Fixed hierarchy support, especially after a sort
- Notes:
 - "classic" hierarchy mode expects hierarchy info in order of the data rows - one row, one hierarchy entry
 - transposed/breaks mode must support visual rows, which do not match the data rows -> hierarchy info in visual order of the data rows + break rows (columns in transposed mode)
 - "visual order", because ALP Area DontSortArrays (using sort index array) is still implied/used when using hierarchy with arrays.
- During entry, data row is not drawn selected even in single-row selection mode
- Fixed column lock divider drawing when lock is not in effect (no divider on the left side of first column)
- Fixed column divider drawing - ALP Area ShowColDivider properly honored
- Fixed picture drawing on Windows when ALP Area WindowsText = 1
- Fixed old bug (from v9.2) in right cell icon drawing (left/right cell icon ID is now 32-bit integer)
- Fixed possible crash when re-initializing "default" area 0
- Fixed re-initialization of area when "default" area 0 is defined (not clearing some properties)
- Fixed many possible crashes when reconfiguring an area displaying breaks
- Fixed bugs related to ALP Area Transposed = 1
 - note: hierarchy and selected rows use visual rows - row selection is always horizontal
 - when using transposed view, you can use negative number as column number to access properties of the virtual columns
- Fixed row number getter when using breaks
- Fixed column dividers drawing when area is scrolled and a column is locked
- Fixed Date popup on Mac when used with 32-bit 4D - month/year popup should work (not to be displayed behind the date popup window)
- Fixed drawing of column dividers when area is wide enough to show all columns
- Fixed possible issues with setting multiple rows/columns/cells at once (especially when using text value)
- Fixed break-processing related problems
 - breaks are deleted when columns are added
 - break row/cell options are not cleared when breaks are deleted

- when processing breaks in fields mode, the break column(s) are not damaged anymore (4D runtime error when fetching the data)
- Fixed use of ALP Area NamedSelection
- Fixed getter/setter for ALP Row UseEllipsis (was missing)
- Fixed use of ALP Column CallbackFnc when using break processing (row out of range)
- Sort works when using: arrays, hierarchy and break processing
- Fixed footer handling when break processing is used and there is TOTAL (break footer for level 0 - displayed in the footer)
- Fixed break row handling - break rows do not change the column widths (it changed them sometimes having effect of “jumping” data on screen, e.g. after a click in column resize)
- Fixed crash when changing number of rows and using break processing and hierarchy (e.g. ALP Area AutoHierarchy = 1)
- Fixed crash when getting ALP Column CallbackFnc
- Added missing setter for ALP Column Ellipsis
- Fixed crash with ALP Column CallbackFnc
- Fixed parsing of format for icon numbers when ALP Column DisplayControl = 4 (draw pictures)
- Fixed row height computation for GDIPlus (running on Windows, ALP Area WindowsText = 0) - visible e.g. with Arial 13
- Fixed pictures in sort dialog (added missing sort direction triangles)
- Fixed a possible crash on quit (Windows)
- Fixed a bug on Windows getting pictures from system (native appearance: header, checkbox) - sometimes it could create empty picture
- Fixed a bug on Windows related to HDC manipulation when creating pictures from system
- Changed Aqua8 pictures (was using 10.6 Aqua – darker blue)
- Fixed a drawing issue when there are many rows hidden (e.g. collapsed hierarchy nodes)

Version 10.0 – Direct data display from a MySQL database

Set a MySQL server as data source and display its data in AreaList Pro v10, with hundreds of formatting options available.

Version 10.0 – Global settings and templates

- Global area settings:
 - Area = 0 : apply to all areas to be created (defining an area “template”)
 - Area = -2 : apply to all existing areas, but not to areas to be created
 - Area = -3 : apply to all existing areas in current process only

- Global column settings:
 - Column = 0 : apply to all columns to be created (defining a column “template”) in either a specific area, or in area 0, -2 or -3 as above
 - Column = -2 : apply to all existing columns in either a specific area, or in area -2 or -3 as above for existing areas
- Global row settings:
 - Row = -2 : apply to all rows in either a specific area, or in area -2 or -3 as above for existing areas

Version 10.0 – New commands

- Move a column (*AL_MoveColumn*)
- Duplicate a column (*AL_DuplicateColumn*)

Version 10.0 – Grouping columns and headers

- Group columns together
- Dragging a group of columns
- Column group header (above the column individual headers or replacing them) – row and cell properties can be used with Row = -3
- Hierarchy row merged into one cell that goes across several columns

Version 10.0 – Area display features

- Zooming an entire area (smaller or larger display, factor 0.1 to 5)
- Display Transposed (rotate rows ↔ columns) – This allows “palette” style interface, with different data types among “rows” (note: “palette” style was already possible in v9 using “grid” mode)
- Multi-level break processing (ala PrintList Pro) including in hierarchical lists
- Many new “appearance” settings: system, automatic, custom pictures or 10 different OSX/Windows standard themes
- Show/hide vertical scrollbar

Version 10.0 – Column features

- Minimum column width for any column(s)
- New properties to enable/disable each column's sorting, dragging or resizing
- “Use ellipsis” settable for headers, specific columns, individual rows, footers, and even individual cells

- Popup entry type is now a column property: 0 = default (depends on data type), 1 = date, 2 = time
- Limiting entry string length (length limit is now a column property)
- Define a formula instead of a callback for a calculated column

Version 10.0 – Other cool features

- CSV format in Copy / Drag from an area
- Named selections as sources
- Entry placeholder support
- ALP Cell Format and ALP Cell Flags (formatting at cell level)
- ALP Row Flags (formatting at row level: font, color, alignment, spacing, etc.)
- Show a longint value as picture in Field or Array Display (e.g. “star rating” display)
- Access row and cell properties for header and footer:
 - Header: row = 0
 - Footer: row = -1

Version 10.0 – Miscellaneous

- Hide row dividers in empty areas (ALP Area ShowRowDividers is now 3-state)
- Custom picture above vertical scrollbar (old AL_SetHeaderOptions) including callback on mouse down or up
- Override row/cell bottom row divider color (set a specific color for the bottom divider of a given row or cell)
- ALP Column EntryAllowReturn: allow/disallow return in specific columns regardless of the area settings (ALP Area EntryAllowReturn)
- Single line headers in “grid” mode (i.e. several lines per row, ALP Area AltHdrRowsInGrid>1)
- Single line footers in “grid” mode (i.e. several lines per row, ALP Area FtrRowsInGrid>1)
- Drag and move a column (physically reorder the column) even when not running in compatibility mode
- New API replacement of AL_SetInterface: appearance + sort indicator properties
- ALP Object RowSelection optimized (using **LONGINT ARRAY FROM SELECTION**)

Upgrading from previous versions

Any v10 license key will activate any v10.x version.

Upgrades from versions 8.x and 9.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