



Bartels AutoEngineer®

Version 7.8

Release Notes

This documentation contains information about the new features introduced with **Bartels AutoEngineer Version 7.8**. Forward compatibility from earlier versions to **Bartels AutoEngineer Version 7.8** is ensured, but not backward compatibility.

Bartels AutoEngineer Version 7.8 Release Notes

Published by: Bartels System GmbH, Munich

Last printing: May 2012

The information contained within the Bartels AutoEngineer publications as well as the products and/or programs described therein are subject to change without notice and should not be construed as a commitment by Bartels System.

Although Bartels System has gone to great effort to verify the integrity of the information provided with the Bartels AutoEngineer publications, these publications could contain technical inaccuracies or typographical errors. Bartels System shall not be liable for errors contained therein or for incidental consequential damages in connection with the furnishing, performance, or use of this material. Bartels System appreciates readers' and/or users' comments in order to improve these publications and/or the products described therein. Changes are periodically made to the information therein. These changes will be incorporated in new editions of the Bartels AutoEngineer publications.

All rights reserved. No part of the Bartels AutoEngineer publications may be reproduced, stored in a retrieval system, translated, transcribed, or transmitted, in any form or by any means manual, electric, electronic, electromagnetic, mechanical, chemical, optical, or otherwise without prior express written permission from Bartels System.

Bartels AutoEngineer®, Bartels Router®, and Bartels Autorouter® are registered trademarks of Bartels System. Bartels User Language™ and Bartels Neural Router™ are trademarks of Bartels System. All other products or services mentioned in this publication are identified by the trademarks or service marks of their respective companies or organizations.

Copyright © 1986-2012 by Oliver Bartels F+E
All Rights Reserved
Printed in Germany

Contents

1	General	5
1.1	Installation	5
1.2	User Interface and General Functions.....	5
1.3	Symbol and Part Libraries	9
2	Schematic Editor	10
2.1	General.....	10
2.2	Symbols, Labels	12
2.3	Connections	16
2.4	Graphics	16
2.5	Text, Attributes	17
2.6	Group Functions.....	18
2.7	Plot Output	19
2.8	Hierarchical Circuit Design	19
3	Packager	20
3.1	Warning Messages.....	20
3.2	Attributes	20
3.3	Logical Library.....	20
4	Layout Editor	21
4.1	General.....	21
4.2	Display, Design Rule Check.....	23
4.3	Parts, Placement.....	25
4.4	Traces, Routing.....	26
4.5	Graphic, Copper Areas.....	29
4.6	Text, Drill	30
4.7	Group Functions.....	31
4.8	Automatic Copper Fill	32
5	CAM Processor	33
5.1	General.....	33
5.2	Control Plot.....	34
6	CAM View	35
6.1	Data Import.....	35
7	Utilities	36
7.1	BSETUP	36
7.2	LISTDDB	36
7.3	LOGLIB	36
8	Bartels User Language	37
8.1	General.....	37
8.2	System Functions.....	37
8.3	BAE User Language Programs	38

1 General

1.1 Installation

Installation Guide

The [Bartels AutoEngineer® Installation Guide](#) (file `inst_en.htm` from the `baedoc` directory of the BAE-CD-ROM) provides detailed **Bartels AutoEngineer** installation instructions for all supported hardware and software platforms.

Configuration Files (Windows)

The BAE **Windows** functions for accessing configuration files have been modified to scan **BAE** sub-directories within user-specific data directories. The configuration file lookup checks the configuration file environment variable, the user-specific BAE application data directory, the general All Users BAE application data directory, and, finally, the BAE programs directory.

During installations/updates, the BAE setup programme copies the window positions and file history configuration files into the user-specific BAE application data directory, the BAE license file and the language-specific message file into the BAE programs directory, and all other files into the general BAE application data directory. During update installations, existing configuration files are moved to their new default directories if necessary. Users can move color table files into their user-specific BAE application data directory. I.e., each user can define and use her/his own color tables.

Keeping the BAE configuration files in BAE application data directories simplifies configuration file transfers between different computers as it allows for complete configuration file data sets to be transferred by copying directories rather than selecting and copying specific file types from the BAE programs directory. The modified BAE configuration file also simplifies the BAE installation into the **Windows** programs directory on **Windows** computers with user access control.

Please note that the real application directory paths may be different in different **Windows** versions. The [Help / Product Info](#) function has been modified to list the full pathnames of all used BAE configuration files.

HighEnd Installation (Windows)

The files required for installing **BAE HighEnd** have been added to the BAE setup program (`setupen.exe`) which is available on the BAE CD and in the BAE update section on the Bartels web site. This allows for the installation of **BAE HighEnd** without the need of having the BAE CD inserted or unpacking an additional file with **BAE HighEnd** files.

1.2 User Interface and General Functions

Display Performance (Windows)

The graphic system performance for displaying patterns has been significantly improved for **Windows** versions with **Desktop Window Manager (DWM)** such as **Windows Vista** and **Windows 7**.

Info Display (Windows)

The tooltip display at the mouse cursor left graphic artefacts on the screen in the BAE **Windows** versions if the information text contained `&` characters. This problem has been fixed.

Window Sizes

The dialogs with scrollable lists and/or reports can now be resized. Dialog size configurations are restored upon subsequent calls.

Please note that window dimensions and window positions were already saved in previous BAE versions. Dialogs which have been automatically resized for displaying short element lists in previous BAE versions may have to be manually resized for conveniently displaying longer element lists.

The [Symbol Browse](#) and [Part Browse](#) dialogs have been modified to automatically adjust their dimensions to achieve the aspect ratios specified through the preview bitmap sizes from `bae.ini` whilst leaving space for required element selection controls.

Toolbar

The toolbar zoom windows were not saved when the saved element matched the last element in the saved element history. This problem has been fixed.

The layout of the top/left toolbar buttons in the BAE **Windows** version has been adapted to the currently activated **Windows** skin. The colors of user-specific letter buttons defined through `TB_ACTBUTN*_STDbae.ini` parameters are not affected. The display mode with selectable colors can be reactivated by attaching an arbitrary letter to the button definition (as in `Fx` instead of `F`). Button definitions starting with an `i` and containing a point character trigger the search for an icon file in the BAE programme and user directories. Button definitions starting with a `b` and containing a point character trigger the search for a bitmap file in the BAE programme and user directories. The button size is calculated based on the current character font size selection. Icons with a size of 16×16 pixels with a 2 pixel wide transparent border are recommended for **Windows** default font size settings.

Command History

The command history which is available through right mouse button clicks on the **H** toolbar button has been modified to insert any command history selection as new last command in the history to allow for subsequent calls to that command through left mouse button clicks.

Mouse Buttons (Windows)

The **STEPUP** and **STEPDOWN** **User Language** programs have been modified to page through the element load history of the current BAE session if no other menu function is active. In the BAE **Windows** version, these programs are assigned to the mouse "thumb" keys, thus providing quick access to the previously loaded element which is useful after `Load Macro` context menu calls.

On default, the display menu can be activated by pressing the middle mouse button or the space bar. The `MMBDISABLE_STD` parameter for deactivating the assignment of the display menu to the middle mouse button has been added to `bae.ini` to make the middle mouse button available for other purposes such as mouse gesture controls.

Mouse Double Click

On default, a double-click of the left mouse button in selection loops of the BAE **Windows** version finishes the selection and displays an undo/redo menu. The `PLoopDBLM_STD` parameter for modifying this behaviour has been added to the `bae.ini` file to allow for left mouse button double-clicks to only terminate the selection loop (default) or to interpret the second click as normal mouse click (e.g., to apply a further rotation step when selecting elements for rotation, etc.).

Coordinate Localization

The `Center/Pan Window` display function has been modified to activate a context menu with the `Jump relative`, `Jump absolute` and `Jump 0,0` functions when clicking the right mouse button. This allows for the viewport to be positioned to a specific coordinate such as an error location from an error report.

Polygon Editing

The `Back` context function for removing the last polygon point during polygon and trace editing interactions has been assigned to the `n` key.

The functions for setting arc center points in polygons and traces have been modified to toggle between `Arc Left` and `Arc Right` when pressing the `m` key.

The toolbar `D` button/function for activating element deletion loops has been modified to support right mouse button clicks for deleting selectable polygon and/or trace corner points rather than complete polygon/trace elements.

The default `s` assignment has been modified to load a list of named dimensions from a file or the clipboard. Each input data line consists of a dimension name and a millimetre dimension value pair. Valid name and value separators are space character, equal sign, comma or semicolon. The loaded values are processed when hitting the `s` key during polygon editing. After selecting the desired dimension, the `Jump Left`, `Jump Right`, `Jump Up`, `Jump Down`, `90 Degree Arc`, `180 Degree Arc (Radius)` and `180 Degree Arc (Diameter)` options can be used to use the selected dimension in `Jump relative` or arc drawing commands. The arc direction is, like with the `c` key function, automatically retrieved from the position of the last drawn segment relative to the mouse pointer position.

Text Fonts

The `standard_s` character font with wide character widths ("M", "W", ...) reduced to normal character widths has been added to the `ged.fnt` font file. The `standard_s` font can be used to avoid small text clearance problems between adjacent characters on copper layers.

Signal Layer Setup (Windows)

Upon startup, the main menu setup utility failed to load signal layer settings from the setup file correctly. This problem has been fixed.

BAE Remote Control

The new **CMDCALL User Language** program for entering and executing a command sequence has been assigned to the `Shift-Ctrl-R` key. The BAE menu function command sequences are listed as key programming sequences in the BAE reference manual. **CMDCALL** is similar to the `Run User Script` with the difference that the BAE window rather than a separate dialog window has the input focus. This allows for external programs to remotely control the BAE through the transmission of key/command sequences.

The `bae.sxp` (standard operations for single elements) and `bae_loop.sxp` (standard operations in input loops) example files for controlling the BAE with the **StrokelIt** mouse gesture program control tool have been added to the `baeconf` directory of the BAE CD-Rom.

Mouse Context Functions

Depending on the **Windows** version and the current menu text size configuration, the right mouse button context menu was occasionally displayed too far from the mouse position when clicking close to the right and/or lower viewport border. This problem has been fixed.

The `Select (Crossing)` function has been made available for right mouse button context function assignments through the mouse button/symbol in the toolbar. `Select (Crossing)` function allows for the selection of elements which are only partially within the group selection frame.

The toolbar mouse symbol dialog box for setting context functions and pick element types has been modified to stay permanently open. Thus, the element types selected for context picks are visible whilst working and can be modified without without having to reactivate the dialog.

Hotkey Display (Windows)

Some shortcut key menu texts were sometimes assigned to wrong (higher-level) menu tree entries upon initial menu display without loaded element and/or after menu text updates after only key programming. This problem has been fixed.

Control Element Appearance

The **Windows** modules have been modified to support **Visual Styles** for a more enhanced display of dialog control elements.

Element References

`References` control elements have been added to the element load and macro placement selection boxes to allow for the name of the first referring element to be appended in parenthesis to element names and/or element comments in the name selection boxes. Ellipsis symbols (`, . . .`) are displayed to indicate the existence of multiple referring elements.

The `ESELREF_STD` parameter for specifying the default reference display mode for project and library element selections has been added to the `bae.ini` file.

Element Preview (Motif)

The element selection dialogs of the `File / Load` submenu function have been modified to include selected element previews when used under **Motif**.

List Dialog Boxes (Windows)

Automatic element list scrolling in list selection boxes with text input fields was only triggered for text inputs less than or equal to the last name in the alphabetically sorted element list. This behaviour has been modified to trigger automatic scrolling to the end of the element name list if the text input is larger than the last name of the element list.

File Utilities

The functions from the **File / File Utilities** menu have been modified to use the last output file name of the current BAE session as default for source file names. This is helpful in functions such as **File View** which is frequently used to load/view a (CAM) file which has just been exported. The last output file name can be set as default in key programming and macro definition call sequence extensions by specifying an empty string.

Report Display Boxes

The **Edit** button for writing the text of the current output window to a file named `projectname_reportid.txt` and automatically loading this file with the application configured to handle `.txt` files has been added to the report output windows. This simplifies text searches in reports and allows for the work to continue in BAE with the report still displayed in the `.txt` application.

Rule System

The rule system memory management has been optimized. Elements with large numbers of rule settings are now loading significantly faster.

File Access Errors

The function for loading DDB file elements has been modified to issue a warning message if the accessed DDB file is write-protected to indicate that subsequent element modifications can not be saved to that file. The file access warning message is issued only when loading the first of multiple elements from the same write-protected DDB file. The `FMODEWARN_STD` parameter for deactivating these warning messages for either productive BAE versions or the BAE Demo version has been added to the `bae.ini` file.

The `FILEERRDISP_STD` parameter for displaying file access warnings not only in the status line but also in a popup window with verification prompt has been added to the `bae.ini` file to ensure that write access failures (due to temporary network access problems) don't go unnoticed.

SQL Databases

The **File / File Utilities / SQL Table Import** function ignored input lines with an empty string in the first field. This behaviour has been changed to ignore input lines only if all fields contain empty string.

Message System (BAE HighEnd)

The `bae.exe` module in **BAE HighEnd** crashed or terminated too early in special cases, thus preventing module switches or data from being transferred between modules. This problem has been fixed.

1.3 Symbol and Part Libraries

New Library Elements

The following elements have been added to the symbol and part libraries supplied with the BAE software:

Library	Type	Elements
ROUTE	Symbol	att_infotext, tag_net_infotext, tag_netarea_infotext, tag_netpin_infotext, tag_sym_infotext

Attribut Symbols

The flag pole graphic areas of the symbols in the `route.ddb` SCM library have been converted to graphic lines. This ensures that connections to those symbols are always in line with the flag poles, regardless of the current plot pen width setting. The display of the irrelevant tag pin names is now suppressed by applying `Move Name` to set the tag pin text size to 0.0.

2 Schematic Editor

2.1 General

File Name Drag & Drop (Windows)

The **Windows** version of the **Schematic Editor** has been modified to support the automatic loading of the schematic sheet configured with the `AUTOLOADSM_STD bae.ini` parameter if a `.ddb` file name is dragged and dropped onto Schematic Editor window.

Clipboard (Windows)

The functions for placing elements and polygon points have been modified to retrieve coordinate pairs from the **Windows** clipboard when pressing the `Ctrl-V` key (**Paste**) while the crosshair-cursor is active. This allows for placement and/or polygon coordinates to be copied and pasted from design data sheets/files (thus also eliminating potential typing errors). The first two numeric values from the clipboard text are interpreted as X and Y coordinates. These coordinates are considered to be in millimetre units unless they have a specific unit suffix such as `mm`, `inch` or `"`. When editing polygons or connections, the `Ctrl-V` function tries to retrieve multiple coordinate pairs from the clipboard text.

Coordinate pair specifications can be prefixed with `:l` for **Arc Left**, `:r` for **Arc Right** or `:n` for **Done + Next**. Other non-numeric text in coordinate specifications is simply ignored.

The **Copy Coordinates** button for copying placed element coordinates as text into the clipboard has been added to the **p** element properties dialog.

Element Save Query

The cancellation of placement operations does not modify the element saved status anymore. This eliminates redundant element status queries and procedures for unchanged elements.

Default Library

The `AUTODEFLIB_SCM` parameter for automatically activating the default SCM library selected through the BAE setup instead of the previously selected library when loading an SCM element has been added to the `bae.ini` file.

Update Library

The **Update Library** function has been modified to use the currently selected project library as default for the source file name selection to eliminate the need for navigating from the project directory to the library directory in cases where the project library rather than the project file needs to be selected.

The **Update Library** has been modified to scan for sub-macros of unreferenced macros in the selected project file. nun ggf. deren Untermakros in der selektierten Projektdatei to ensure that pin markers from the selected source file are updated even if the referring symbols and/or labels are not in the source library.

The **File / Update Library / All Sheets** function sometimes failed to process macros which were exclusively used on the first project scm sheet. This problem has been fixed.

Orphant Macro Search

The **Mark Unreferenced** button for selecting unused (orphant) library macros for subsequent DDB file element delete procedures has been added to the element selection boxes of the **File / Library Utilities / Delete Elements** submenu. This feature significantly simplifies DDB file cleanups.

Logical Library

The **File / Library Utilities / Delete Elements / Logical Library** function has been converted to a submenu with the **Single DDB File** and **Project File Tree** functions. The **Single DDB File** function deletes logical library elements from a single selectable DDB file. The **Project File Tree** function can be used to remove logical definitions specified through names or name patterns from all DDB files within a selected directory tree. This is useful if an erroneous logical library definition needs to be globally removed from DDB project file groups without affecting any other definitions with project-specific modifications.

Schematic Editor Startup

The **Schematic Editor** has been modified to support startup with a command line such as

```
> bae scm DDB-filename planname ULC:commandsequence ↵
```

or

```
> bae scm DDB-filename planname ULC_QUIT:commandsequence ↵
```

for loading a specific schematic plan from a DDB file into the **Schematic Editor** and executing the specified command sequence. This allows for the implementation of batches such as fully automated direct PDF output. **ULC_QUIT** causes the **Schematic Editor** to close down after executing the command sequence.

EDIF Import

The **File / Import/Export / EDIF Data Import** function for importing schematic plan data in EDIF format has been modified to support partially negated name texts. The **EDFFULLNEG_SCM** *bae.ini* parameter which was previously used to control this behaviour has become obsolete and will be ignored.

The **EDFRENAME_SCM** parameter for specifying whether name texts imported through the **rename** EDIF command should be left unchanged or converted to a portable format (special characters substituted with underscores) has been added to *bae.ini*. On default, imported name texts are converted to the portable name format.

The **EDIF Data Import** function has been modified to translate layout name changes as specified with the **instanceBackAnnotate** EDIF command to BAE **\$rpname** attribute assignments.

If the the name prefix for a symbol macro contains a questionmark, then this questionmark is now substituted with an asterisk name pattern character, and the numeric part of the symbol name pattern specified through the **EDFNAMEPAT_SCM** *bae.ini* parameter is omitted.

Labels in hierarchical schematic plans with **port*** name pattern macros are now imported as module ports with the pre-defined **modul port** label macro. Previously, such labels were imported as net name labels and caused problems when repeatedly referenced in hierarchical blocks.

2.2 Symbols, Labels

Symbol Search

Synthetically generated tag symbol names for net areas have been removed from the list which is displayed in the **View / Find Symbol / Search in List** dialog.

A column for the SCM sheet element comment has been added to the symbol list.

The BAE **Windows** version has been modified to use a listview control with variable column width and sortable column content for the symbol list display.

Label Sheet Search (BAE HighEnd)

The **Label Sheet New Window** function for loading label sheets into new **Schematic Editor** windows has been added to the context menu for labels. This allows for source and destination labels of multi-sheet nets to be displayed simultaneously.

The **Sheet Filter** button for restricting the net name listing to the nets of a selectable project sheet has been added to the net name selection menu of the **Symbols / Add Label** function. This simplifies net name selections in cases where the target sheet is known.

Symbol Movement

The functions for moving symbols and labels have been modified to display symbol/label attribute values instead of symbol/label macro attribute name texts. With activated **Keep Text Positions** option moved name and attribute texts are displayed at their modified positions during manual symbol/label placement operations.

Symbol Attribute Texts

The **Names/Attributes** option has been added to the toolbar mouse symbol for selecting the pick element type. With **Names/Attributes** activated, a **Move Name** and/or **Move Attribute** call instead of **Move Symbol/Label** is triggered with **Shift** key symbol context picks and in **B** toolbar button loops if a symbol name or symbol attribute text is placed at the pick position.

Label Placement

The automatic label placement functions from the **Symbols / Symbol Edit Functions** have been modified to skip direct **Add Connection** calls as these cause disruptions to the placement process. Connections to automatically placed labels can usually be created more efficiently using the **Draw Pattern** function.

Module Ports

The **View / Query Element** function (**Ctrl** key) has been modified to display module ports on hierarchical schematic sheets correctly instead of listing them as labels with questionmark prefix.

The **p** key dialog has been modified to allow for module port data to be edited.

A label type assignment feature for switching between net label and module port has been added to the **p** dialog. This makes it easier to convert top level SCM blocks with label connections to other SCM sheets into hierarchical blocks with module port connections.

The module port macro name is not pre-defined (**port**) anymore. Module port macro names can now be assigned through the right mouse button context menu in the port placement functions and/or through the new **Port Label Macro** parameter in **Settings / Settings**. This allows for different graphic symbols/labels to be used for different module port types such as input and output ports.

Group Attribute Assignment

The **Edit / Other Functions / Set Group Attributes** function has been modified to activate a user confirmation prompt for overriding attribute value assignment locks. For locked attributes with default value list, default value assignments are accepted without verification. The attribute value input dialog provides a **Browse** for displaying attribute default value selection lists.

Symbol Attribute Tables

The **CSV Export** and **CSV Import** functions for exporting and importing symbol attributes data in CSV format with various dialog options for selecting outputs and file formats have been added to the **Symbols / Other Functions / Set Attributes** menu. This allows for exported project attribute values to be edited using an external spreadsheet application and then import the modified attribute values back into the project.

The attribute table import function has been modified to support multiple comma-separated or space-separated symbol names in the symbol name field/cell. This allows for multiple symbols with identical attribute values to be specified in a single input line/row.

Symbol Attribute Replacement

The **Symbols / Other Functions / Set Attributes / Replace Attributes** function has been modified to list a symbol attribute modification report at the end.

Symbol Attribute Cleanup

The **Symbols / Other Functions / Set Attributes / Attribute Cleanup** function for removing attribute values without corresponding symbol attribute text definitions from selectable project files. This allows for automatic project file attribute values cleanups after removing attribute definitions from library symbols. The **Attribute Cleanup** functions issues a report of the removed attribute values.

Symbol Attribute Default Values

The **Symbols / Other Functions / Set Attributes / Attribute Initialization** function for assigning default symbol attribute values to all or unset symbol attributes in a selectable project file has been implemented to allow for project attributes to be automatically set to new or changed default attribute values.

Header Symbol Attributes

The **Set Frame Attributes** function reset dialog attribute changes in the dialog when releasing locked attributes. This problem has been fixed.

Symbol Browser

The **Symbols / Other Functions / Symbol Browse** function has been modified to include the symbol name pattern of the selected symbol in the symbol preview.

Symbol Database

The `SSELSBOXID_SCM` parameter for configuring a common position id for the symbol database dialog boxes has been added to `bae.ini`. This ensures that the symbol selection window is displayed at the same position as the top level selection window if the top level window has been moved inbetween symbol selections.

The full selection path has been added to the end of symbol database search result lists, thus providing an indication of the selected category. The `SSELPDOC_SCM` parameter for restricting the display to the lowest selection level has been added to `bae.ini`.

The `close` control element for closing the symbol selection dialog during the placement of a selected symbol has been added to the symbol database dialog boxes. This feature can be used to reliably prevent the symbol selection dialog window from concealing the workarea during placement operations (on single-screen workstations). The `SSELCLOSE_SCM` parameter for setting a default for the `close` behaviour has been added to `bae.ini`.

The `ID Replacement` symbol database function has been modified to list the names of changed symbols and variants together with the number of modifications.

The symbol selection in the BAE **Windows** versions is now displayed in a listview control with sortable width-adjustable attribute columns.

The `SSELFULLH_SCM` parameter for controlling whether column headers are to be considered for column width calculations and the `SSELATRIM_SCM` parameter for automatically adjusting column widths upon dialog activation have been added to `bae.ini`.

Symbol Documentation

The `Settings / Rule Attachment / Plan` function has been modified to activate an edit field with `Browse` button rather than a file selection dialog when assigning the `part_docu` string predicate to the part macro documentation. This allows for the specification of web links or document query scripts. The function for displaying part macro documentation automatically activates the web browser which is configured as handler for `.htm` files when referring to a part macro documentation without or unknown file name extension.

Cross Highlight (BAE HighEnd)

The `Layout Editor` functions for placing and/or moving layout parts in **BAE HighEnd** have been modified to highlight corresponding schematic symbols in open `Schematic Editor` project windows by filling the symbol macro boundary areas with a grid pattern. This allows for distinguishing between unplaced and placed or moved parts on the schematic sheet.

Tag Symbol Info

The current element report which can be displayed through the `F1` key has been modified to include tag symbol mode and tag pin type info on SCM symbol macro level.

Tag Symbol Check

The tag symbol pin destination selection has been modified to reject the pin's own tag symbol to avoid invalid tag assignments (e.g., when accidentally double-clicking a tag symbol).

The `Utilities / Report` function has been modified to produce a separate entry for open tag symbol pins instead of listing them as open contact areas.

A separate section for open and self referencing tag pins including information about the types of the tag symbol pin destination elements has been introduced to the reference listing which is produced by the `Symbols / Other Functions / SCM Cross Reference` function.

Library Macro Check

The **File / Library Utilities / Library Check** function has been added to allow for the generation of a listing of standard library macros with a different modification date than the corresponding macros of the currently loaded project element. The report dialog provides a **Update Library** button for carrying out a library update with the standard library without further queries.

The **MACLIBCHECK_STD** parameter for triggering automated **Library Check** calls when loading elements has been added to the **bae.ini** file.

Library Documentation

The **Set Default Values** option for assigning the default attribute values defined on symbol macro level to the symbols on the generated documentary sheets has been added to the **Files / Library Utilities / Library Documentation** function. I.e., the **Set Default Values** option can be used to ensure that the symbols are displayed with their default positioning parameters in the generated documentation.

2.3 Connections

Connection Placement

Placing a connection end point in edit direction onto an antenna segment finishes now the editing of the current connection and starts a new connections.

Net Colors

The `Color/Hatch Nets` function for coloring nets and/or assigning net display patterns has been added to the `View` menu.

The **BAE HighEnd** has been modified to transfer `Color/Hatch Nets` settings to **Layout Editor** sessions for the same project.

The `EPS/PDF Output` function has been modified to consider the `Color/Hatch Nets` settings.

Connection Splitting

The `Split Mode Connections` parameter with different options for processing connections which are drawn over multiple symbol pins has been added to the `Settings / Settings` dialog. The `No Connection Split` split mode option leaves the multi-pin connection segment unchanged. The `Connection Split` option automatically splits connection segments which are drawn over an even number of symbol pins. The `Only at 2 Pin Symbols` option only split connection segments which are drawn over both pins of 2-pin symbols. The `Split Mode Symbols` parameter with corresponding options has been added to control the connection segment split mode for symbols which are placed onto existing connections segments. The split mode options can be used to significantly simplify the process of connecting inline placed symbols.

Buspick

The pick sensitivity for bus segments has been extended from the bus center line to the whole bus width to simplify the selection of busses when working with very high zoom factors.

Bus Taps

The system has been modified to derive bustap plot visibility settings from bus segment plot visibility settings only for bustaps without connections. Bustaps with connections are only visible in plot outputs if at least one directly connected bus tap connection segment has plot visibility.

Diagonally placed bus taps (imported from foreign systems) sometimes caused bus tap connection end positioning problems when moving the bus taps and/or modifying bus tap connections. These problems have been fixed.

2.4 Graphics

Graphic Corner Edit Loops

The `Done + Next` function for finishing the current graphic element and starting a new graphic element of the same type has been added to the right mouse button context menus of the functions for editing graphic corner points.

The `:mr:s12` command sequence can be used to assign the `Done + Next` context function to a key.

Each graphic created with the `:mr:s12` command sequence is assigned to its own undo step to ensure that only the last graphic is discarded when cancelling the last graphic input.

2.5 Text, Attributes

Negation Lines

The `\ /` character sequence for assigning an overbar to the following character is now supported for texts and reference names to indicate a negation as in `R/\ /W` where the `w` in `R/W` is negated. The string to be negated can alternatively be enclosed with `\ (` and `\)` as in `READ/\ (WRITE\)`.

The `bae.ini` parameters for specifying character aspect ratios and text height scaling factors for the `AutoCAD/DXF Output` function with DXF fonts specified through the `DXFFONT_SCM` parameter must match the selected DXF font characteristics to ensure that character overbars are placed correctly in the DXF output.

Text Alignment

The `Text Mode` and `Origin Text` batch steps of the `File / Library Utilities / Symbol Edit Batch` have been modified to allow for optional text center alignments.

Multiline Texts

The `MTEXTORIG_STDbae.ini` parameter for re-configuring multiline text pick points from bottom left to top left was not correctly adhered to by the `p` key dialog for editing multiline texts. This problem has been fixed.

Text Search

The `Search Text` function for searching text elements has been added to the `View / Find Symbol` submenu. `Search Text` lists the texts placed on the currently loaded element and triggers a `Zoom Window` to the position of the selected text when double-clicking an entry in the text list.

Symbol Edit Batch

The `Text(s) Group Name` batch step for assigning specific attribute texts to a group name has been added to the `File / Library Utilities / Symbol Edit Batch` function.

External Document References

The `Text / Other Functions / Link to External File` function has been modified to support file links with space characters in the file and/or directory name.

Symbol Attribute Assignments

The `Settings / Rule Attachment / Net Area` function has been renamed to `Net/Part Area`. Areas created with this function can be tag symbol pin targets for the new `Part Area Tag` type for transferring tag symbol attributes to all symbols which have their origin positioned within the assigned area. This allows for the assignment of symbols within a specific area to a block name attribute for subsequent selection of the corresponding parts in the layout.

Clipboard (Windows)

The `Ctrl-V` copy function has been modified to activate a symbol attribute assignment loop if the clipboard contains attribute assignment texts such as:

```
$val=100K  
$plname=r75  
$id=117 563
```

2.6 Group Functions

Clipboard (Windows)

The **Copy Coordinates** function for copying the coordinates of the group-selected connections and graphic polygons as text to the **Windows** clipboard has been added to the **Edit** menu. The coordinate text format is identical to the format of the **Ctrl-V** function. The **Copy Coordinates** function can be used to transfer a logo consisting of different areas via the clipboard to a different schematic hierarchy level or even into the **Layout Editor** (separate copy/paste steps for lines and filled areas may be required).

The **Copy Coordinates** can be called through the **Shift-Ctrl-C** key.

Group Report

The **Edit / Other Functions / Group Report** function for generating a selected group element report with selection counts and listings of group-selected symbols, labels and texts has been added.

Group Selection

The **Edit / Group Polygon** function has been modified to provide the **Select Polygon** and **Select Crossing** options prior to the selection of the first selection area corner point. **Select Polygon** allows for the selection of an existing polygon as group selection area. **Select Crossing** allows for the selection of elements which are only partially within the group selection area.

The **Texts** option for selecting and/or deselecting text elements through text string specifications has been added to the **Edit / Other Functions / Select/Deselect / Name** functions.

Negated options for macro types have been added to the **Edit / Other Functions / Select/Deselect / Macro** functions to allow for the selection/deselection of macros which don't match the selected name and/or name pattern. This provides additional group selection functions such as the selection of parts which are not placed with the preferred layout package type.

The name selection boxes of the **Edit / Other Functions / Select/Deselect / Name** function have been modified to list only names of elements which don't have the desired group status. This feature doubles as query function for selected and/or unselected group elements.

Group Element Rotation

Functions for left and right rotation and for setting the rotation angle of group selected symbols, labels and texts have been assigned to the **⌘** and **⌘** keys if those keys are pressed while no other function is active.

Group Loading

The **Edit / Other Functions / Load without Hierarchy / Sheet** has been modified to preserve text placement symbol attribute values instead of substituting them with attribute names.

DXF Export

The **Group Elements Only** option for restricting DXF outputs to group-selected elements has been added to the **AutoCAD/DXF Output** function.

2.7 Plot Output

EPS/PDF Output

The `Symbol Attribute Annotations` option of the `EPS/PDF Output` function has been modified to export symbol PDF layer name settings also with symbol attribute annotations. This ensures that the visibility of symbol attribute annotations changes with the visibility of their symbols when fading in and/or fading out layers in **Acrobat Reader**.

The `Initial Display Acrobat Reader` parameter can now be assigned to PDF output layers of PDF batches with element to PDF layer assignments. This allows for the creation of PDF documents with invisible unplaced symbols which can be faded in if required.

Generic Output (Windows)

The `Generic/Windows Printer Output` function has been modified to allow for the specification of a printer name after the layer selection in macros and key programming sequences. The name `standard` can be used to select the **Windows** default printer.

The `all` text can be added to the call sequence to trigger the output of all schematic sheets of the project.

2.8 Hierarchical Circuit Design

Net Highlight (BAE HighEnd)

Net highlights for local nets and module ports of hierarchical schematic blocks were transferred to correspondingly named nets and module ports of other sub-blocks when loading different project sheets. This problem has been fixed.

3 Packager

3.1 Warning Messages

Warning Messages

The **Packager** messaging system has been modified to display warnings about potential `net internal` problems for parts consisting of more than 32 sub-parts only if these sub-parts contain `net internal` commands.

3.2 Attributes

Placement Status

The `$noplc` placement status attribute value setting for base variants was not considered when packing multiple symbols from variants with unassigned `$noplc` values. This problem has been fixed.

3.3 Logical Library

Logical Library Cleanup

The `Cleanup Mode` option for removing unused logical library definitions from the project file after successfully running the **Packager** has been added to the **Packager Settings** menu.

4 Layout Editor

4.1 General

File Name Drag & Drop (Windows)

The **Windows** version of the **Layout Editor** has been modified to support the automatic loading of the layout element configured with the `AUTOLOADLM_STD bae.ini` parameter if a `.ddb` file name is dragged and dropped onto **Layout Editor** window.

Trace and Area Corner Edit Loops

The **Done + Next** function for finishing the current element and starting a new element of the same type has been added to the right mouse button context menus of the functions for editing trace and area corner points.

The `:mr:s12` command sequence can be used to assign the **Done + Next** context function to a key.

Each graphic created with the `:mr:s12` command sequence is assigned to its own undo step to ensure that only the last graphic is discarded when cancelling the last graphic input.

Element Properties

The control elements for changing the coordinate units in the **p** element properties dialog were not working. This problem has been fixed.

Clipboard (Windows)

The functions for placing elements and polygon points have been modified to retrieve coordinate pairs from the **Windows** clipboard when pressing the **Ctrl-V** key (**Paste**) while the crosshair-cursor is active. This allows for placement and/or polygon coordinates to be copied and pasted from design data sheets/files (thus also eliminating potential typing errors). The first two numeric values from the clipboard text are interpreted as X and Y coordinates. These coordinates are considered to be in millimetre units unless they have a specific unit suffix such as `mm`, `inch` or `"`. When editing polygons or traces, the **Ctrl-V** function tries to retrieve multiple coordinate pairs from the clipboard text.

Coordinate pair specifications can be prefixed with `:l` for **Arc Left**, `:r` for **Arc Right** or `:n` for **Done + Next**. Other non-numeric text in coordinate specifications is simply ignored.

The **Copy Coordinates** button for copying placed element coordinates as text into the clipboard has been added to the **p** element properties dialog.

Element Save Query

The cancellation of placement operations does not modify the element saved status anymore. This eliminates redundant element status queries and procedures for unchanged elements.

Default Library

The `AUTODEFLIB_GED` parameter for automatically activating the default layout library selected through the BAE setup instead of the previously selected library when loading a layout element has been added to the `bae.ini` file.

Update Library

The **Update Library** function has been modified to use the currently selected project library as default for the source file name selection to eliminate the need for navigating from the project directory to the library directory in cases where the project library rather than the project file needs to be selected.

The **Update Library** has been modified to scan for sub-macros of unreferenced macros in the selected project file. nun ggf. deren Unter makros in der selektierten Projektdatei to ensure that padstacks and/or pads from the selected source file are updated even if the referring parts and/or padstacks are not in the source library.

Orphant Macro Search

The **Mark Unreferenced** button for selecting unused (orphant) library macros for subsequent DDB file element delete procedures has been added to the element selection boxes of the **File / Library Utilities / Delete Elements** submenu. This feature significantly simplifies DDB file cleanups.

The **File / Delete Element / Padstack** function allowed for the deletion of project via padstacks which are not defined in the layout standard via list. This problem has been fixed.

Layout Editor Startup

The **Layout Editor** has been modified to support startup with a command line such as

```
> bae ged DDB-filename layoutname partname ↵
```

for loading a specific layout from a DDB file into the **Layout Editor** and automatically trigger a **Zoom Window** call to zoom to the position of the specified part.

In addition to that, the **Layout Editor** has been modified to support startup with a command line such as

```
> bae ged DDB-filename layoutname ULC:commandsequence ↵
```

or

```
> bae ged DDB-filename layoutname ULC_QUIT:commandsequence ↵
```

for loading a specific layout from a DDB file into the **Layout Editor** and executing the specified command sequence. This allows for the implementation of batches such as fully automated direct PDF output. **ULC_QUIT** causes the **Layout Editor** to close down after executing the command sequence.

An asterisk character (*) can be used for the layout name in the above **Layout Editor** startup commands to refer to the default layout element name configured through the BAE setup.

Documentary Layers

The BAE setup has been modified to allow for the configuration of up to 12 documentary layers (with side configurations where applicable) for direct access through top level layer selection menus to simplify frequently applied documentary layer selections in functions such as **Add Text** OR **Add Keep Out Area**.

4.2 Display, Design Rule Check

Color Palette

The `Fixed` and `Glued` entries have been added to the special layers of the layout color palette. Only the display pattern selection (but not the color setting) for these special layers is used for the display configuration. In cases where different display patterns are assigned to an element, net display patterns have the highest priority, followed by glue display patterns, fixed display patterns and layer display patterns.

The `Color Save` parameter with the `Color Table Name` and `Colors to Element` options has been added to the `View / Settings` dialog. `Color Table Name` is the default option for saving the name of the last loaded color table with the project. `Colors to Element` saves the current color settings with the project to allow for those settings to be kept through module changes or when loading the project on a different BAE installation where the original color tables are not available.

Airline Display

The `Mincon Color` parameter with the `Unroutes Color` and `Layer Color` options has been added to the `View / Settings` dialog. `Unroutes Color` is the default option for displaying airlines using the color table entry for unroutes. The `Layer Color` option displays the airlines using layer-specific colors to indicate the layers on which airlines connect to pins and to allow for layer-specific airlines to be faded out with the layer. In `Layer Color` mode, the functionality of the `Unroutes` color setting is reduced to fading out all airlines.

The `Mincon Clipping` parameter for suppressing the display of airlines with both start and end point outside the current viewport has been added to the `Unroutes Clipping` airline display option. This can be helpful when manually routing parts in the center of an otherwise unrouted layout.

Solder Side View

An Y-axis mirror view option has been added to the default X-axis mirror view option for the `Solder Side View` mode of the `View / Settings` dialog.

Element Data Query

The `Ctrl` key short info query function for elements at the mouse pointer has been extended to display element glueing/anchoring information.

Net Highlight

The `Reset Highlight/Color` function for resetting net highlights and net color assignments has been added to the right mouse button context menu.

Info Texte

The system has been modified to display part and/or net `$info` attribute values at the graphic cursor when processing and/or querying elements. This allows for the definition of part-specific and net-specific application notes in the `Schematic Editor` and/or in the logical libraries for automatic display during interactive part placement and/or net routing operations in the `Layout Editor`. New tag symbols for setting and/or assigning `$info` attributes have been added to the `route.ddb` SCM symbol library.

Item Browser

The **Item Browser** function for activating a modal dialog for fading in and/or fading out selectable element and/or polygon types of different hierarchy levels has been added to the **Utilities** menu. This feature provides useful visual aids for checking whether certain polygon types are placed on layout, part, padstack or pad level, distinguishing between closed documentary lines and keepout areas on the same layer, etc.

Elements which are faded out can not be manually picked and are excluded from the **Corner Scan** and group selections which are restricted to visual elements.

Faded out element types are automatically faded in when closing the **Item Browser** dialog. The **Save** button can be used to save the current **Item Browser** settings. The **Load** button can be used to load a previously saved **Item Browser** configuration.

The **Item Browser** settings are now considered when starting **File / Import/Export / EPS/PDF Output** and processing batches with current parameter settings. I.e., elements which are currently faded-out from display are not included in outputs anymore.

DRC Error List

The error list of the **Utilities / DRC Error List / DRC Error List** function has been extended to short circuit group numbers for short circuit nets. The highlighted short-circuit area is now restricted to the bounding rectangle of the short-circuit pins rather than the bounding rectangle of the complete short-circuit net. This simplifies the identification of short-circuits for nets with many pins spread over the layout.

The DRC error list has been modified to include net names, net types and net minimum clearances for traces and vias to simplify error tracking for design rule errors caused by non-default net parameters.

The DRC error listing has been modified to include the classes affected by errors between documentary layer keepout errors with class assignments.

The **ERRLBOXW_GED** parameter for setting (increasing) the width of the DRC error list dialog box has been added to **bae.ini**.

The **Enter** key has been modified to trigger an error acceptance toggle if the error dialog has focus and the **Toggle Acceptance** option is active. This allows for error list scrolling and error acceptance toggling through keyboard operation.

DRC Blocks (BAE HighEnd)

The DRC functions have been modified to consider part and/or padstack level DRC blocks (specified through **drdblkc** integer plan predicates) for parts and/or vias without specific DRC block assignments. This allows for the specification of reduced BGA pin distances on part macro level rather than through repeated DRC block assignments with each BGA part placement.

HF DRC (BAE HighEnd)

In multi-core processor systems, the **Utilities / HF DRC** function lost DRC error information after finishing the DRC scan. This problem has been fixed.

Variants

The **Variant Island Report** button for generating a report for all variants with variant-specific lists of nets which are disconnected islands due to **\$noplc** (unplaced) attribute value assignments to all net parts/pins has been added to the variant selection dialog of the **Settings / Variants** function.

4.3 Parts, Placement

Part Data Query

The info field of the **p** key dialog for net list parts has been modified to display drill diameters with part pins.

The **Settings / Rule Attachment / Plan** function has been modified to activate an edit field with **Browse** button rather than a file selection dialog when assigning the `part_docu` string predicate to the part macro documentation. This allows for the specification of web links or document query scripts. The function for displaying part macro documentation automatically activates the web browser which is configured as handler for `.htm` files when referring to a part macro documentation without or unknown file name extension.

Part Renaming

The **Netlist Part Name** control element for selecting either the **Netlist Part Name** (default) or the **Change Part Name** function for part names changes has been added to the **p** key dialog for net list parts.

Part Rerouting

The part rerouting function has been modified to consider the rerouting of traces to the moved part as separate undo/redo step. This allows for undesired trace rerouting results to be reverted without having to repeat the part placement operation.

Part Name Movements

The **Name Move to Attribute Move** layout batch step has been added to the **File / Library Utilities / Layout Library Edit Batch** function. **Name Move to Attribute Move** converts **Move Name** modifications of names and attributes into pure `$` move operations on selectable documentary layers according to **Move Attribute**. This allows for new attribute name texts on **LOGICAL** documentary layers to be de-coupled from previous move name operations which would otherwise relate to all texts starting with `$`.

Part List Output

The **File / Import/Export / Part List Output / BAE Format** function has been modified to include entries for the variant name and variant number in the header of the part list output file if the layout variant selected for output is different from the base variant.

The **Edit Output File** button for automatically loading generated part list files into a text editor (for control and/or printout) has been added to the dialog of the **File / Import/Export / Part List Output** function.

Wrong Part Types

The layout system has been modified to preserve netlist references for parts which are placed with the wrong layout part macro. The DRC displays pin short-circuits and pin airlines for parts which are placed with the wrong package type and processes the part attributes. Both online DRC and **Batch DRC** have been modified to indicate part type errors with a dotted line around the part macro element borders. The **DRC Error List** has been modified to allow for part type errors to be marked as accepted.

Part DRC

DRC errors caused by part placements violating board side preferences are now displayed using hatch areas instead of dashed outlines.

The **Pins** button for assigning net types for the net group DRC to unconnected part pins has been added to the **p** key dialog.

Padstack Generation

The `GLMSPSIG_GED` parameter for triggering a solder layer query instead of automatically generating a signal layer pad when generating SMD padstacks without SMD name extensions has been added to the `bae.ini` file.

Library Macro Check

The `File / Library Utilities / Library Check` function has been added to allow for the generation of a listing of standard library macros with a different modification date than the corresponding macros of the currently loaded project element. The report dialog provides a `Update Library` button for carrying out a library update with the standard library without further queries.

The `MACLIBCHECK_STD` parameter for triggering automated `Library Check` calls when loading elements has been added to the `bae.ini` file.

4.4 Traces, Routing

Point to Point Traces

The `Point to Point Trace` function for activating a simple point-to-point line router has been added to the `Traces` menu. `Point to Point Trace` automatically connects two selectable points on the same layer with a trace. If a net list pin with exactly one unconnected net pin is placed at the selected start position, then the unconnected net pin is automatically selected as end position. If no single routing layer can be determined through the start and end point selections, then the routing is carried out on the currently selected pick preference layer.

Trace Corner Movement

The `Move Corner` function from the `Traces` menu has been modified to save the relative coordinates of the move corner operation for subsequent move corner operations with the new `Repeat Corner Move` function. The `Repeat Corner Move` itself runs in a corner selection loop for speedy trace bundle modifications.

Trace Segment Movement

The `Move Segment` function from the `Traces` menu has been modified to cache the segment movement parameters (movement vector, layer change and trace width change) for subsequent segment move operations with the `Repeat Segment Move` function. With `Repeat Segment Move` applied through a segment selection loop, this simplifies certain tasks such as uniformly moving bundles of parallel trace segments.

Trace Segment Cross

The `Cross Point` function which can be activated through the `x` key while editing traces has been modified. The system automatically extends trace segment pairs to their cross-point if `Cross Point` is activated before setting any trace corner point and the selected (non-parallel) trace segments are on the same layer. This simplifies the completion of pre-routed angle-free bus structures.

Trace Length Query

The `Trace Layer Fade In` control element has been added to the trace length report dialogs of the `Traces / Other Functions / Trace Length / Length Query` submenu to control whether or not all trace layers of a selected net in the `Zoom Window` dialog are automatically faded in.

The `p` key dialog for traces has been modified to consider layer change net length increases (according to layer stack definitions) when calculating and displaying total net trace lengths in **BAE HighEnd**.

Double Trace Segments

The function for finding trace segments which are placed on top of each other has been added to the `Utilities` menu. DRC error markers are placed at the trace segment overlap positions. It is recommended to apply this function prior to net length queries to prevent overlapping trace segments from being added to net lengths.

Pin Coordinate Snap

The pin coordinate snap pick performance of the **p** key has been significantly improved for parts with large numbers of pins.

Trace Width Derivation

The standard trace width instead of the pin-specific net list trace width was sometimes assigned if drill holes were activated in the color table and the first corner point of a new trace was placed exactly on the drill coordinate of a drilled pin. This problem has been fixed.

Fixed Traces

The **Add Fixed Trace** function for manually generating fixed traces has been added to the **Traces** menu.

Trace Merging

The **TRCLMERGEQRY_GED** and **TRCPMERGEQRY_GED** parameters for specifying trace merge modes on layout and/or part level have been added to **bae.ini**. These settings control whether the **Add Trace** and **Move Corner** functions are supposed to merge traces either automatically or upon user confirmation the start or end point of the edited trace is set onto an antenna end point. This simplifies further trace editing as it allows for in-line trace segments to be merged. The default settings are automatic trace merging on layout level and trace merging upon user confirmation on part level.

Trace Insertion

A confirmation prompt for inserting a new trace into an existing trace is now activated if the start and end point of the new trace are selected to be on an existing trace on the same layer and the new trace width matches the width of the existing trace. The inserted trace is automatically fixed if the existing trace is fixed. This simplifies the generation of structures for evading vias as the trace insertion is not bound by the original edit/angle direction settings. The **TRCJOINQRY_GED** parameter activating direct trace insertions without confirmation prompts has been added to **bae.ini**.

Trace Conflicts

The functions for manually editing traces have been extended to activate a menu with the **Ignore Conflict Traces**, **Delete Conflict Traces**, **Delete Conflict Segments** and **Cut Conflict Segments** options if the new trace violates clearance distances to other traces/vias and/or causes short circuits. The default **Ignore Conflict Traces** option leaves everything unchanged. **Delete Conflict Traces** fully deletes existing traces which are causing DRC conflicts with the new trace. **Delete Conflict Segments** deletes existing trace segments which are causing DRC conflicts with the new trace. **Cut Conflict Segments** cuts and deletes existing trace segments to fix the DRC conflicts. The functions for deleting colliding traces/segments in order to reroute them from scratch simplify the manual rerouting of colliding traces since the **airline/unroutes** display provides a better manual routing guidance than the short-circuit highlight. New options in **Settings / Settings** and the **TRCCOLLQRY_GED** parameter in **bae.ini** allow for the selection of a default function for fixing trace collisions and the deactivation of the trace conflict function menu.

Trace Edit Signal Layer Display

Signal layers which are automatically faded in due to layer changes are now automatically faded out again upon subsequent layer changes.

Parallel Traces

The **Traces / Other Functions / Parallel Traces / Other Layer** function for generating trace copies on alternative layers has been modified to support the selection of **All Layers** and **Mid. Layers**. The prompt for an additional alternative layer is skipped with **All Layers** and/or **Mid. Layers** selections, and the trace copy is automatically placed on those signal layers where they don't cause any DRC errors. This feature is useful to avoid problems with optical alternative layer inspections for traces running through very narrowly placed pins and/or vias.

Trace Area Generation

The functions from the **Traces / Other Functions / Trace to Area** submenu have been modified to support traces with arcs and to round outside polygon corners to ensure that the generated polygons are parallel and equidistant to the traces.

Via Data Query

The **p** key dialog for vias has been modified to display the drill diameter and the drill class for the selected via padstack macro.

Via Placement

The **glued** option has been added to the parameter dialogs of the **Place Vias** and **Dot Fill Area** functions from the **Traces / Via Functions** submenu to allow for the direct placement of glued vias.

The **Jump relative** context menu function in **Place Vias** loops has been modified to pick a placement coordinate relative to the last placed via rather than the layout origin.

Via Net Assignment

The net coloring and interactive net pick functions have been modified to consider vias with net assignments for copper fill procedures even if these vias are not yet connected to their designated nets.

Teardrop Generation

Teardrop generation with a **Relative Distance** parameter setting of less than 1.0 sometimes created invalid traces for pad/trace combinations with large diameter differences. This problem has been fixed.

Unroutes List

The **Unroutes List** function from the **Traces / Other Functions / Unroutes Report** dialog failed to autozoom to the selected (double-clicked) airline list entry if the airline is exactly horizontal or vertical. This problem has been fixed.

Net Highlight (Windows)

A left mouse button double-click in the net selection loops of the functions for highlighting and/or coloring nets used to cancel the net highlight function and activate the **Undo/Redo** menu. This behaviour has been changed. Left mouse button double-clicks are now ignored to allow for quick net highlight status toggles without terminating the net highlight functions.

The **NETCOLSAVE_LAY** and **NETVISSAVE_LAY** parameters for automatically saving net color assignments and net visibility modes with the layout have been added to **bae.ini**. The net color assignments and net visibility data set are stored in the project file with the layout name and will be automatically loaded when the layout is subsequently re-loaded.

Net List Assistant

The **L** and **S** buttons for loading and/or saving data sets with airline visibility configurations and/or net highlight settings have been added to the airline visibility controls of the net list assistant dialog.

The **Return** key has been modified to trigger the mouse double-click action if the net name selection box has focus. This allows for scrolling the net list and triggering the default action for the selected net through keyboard operation.

Net List Output

The **Connected Constructive Pins** mode has been added to the **Pin Output** option of the **Traces / Net List Utilities / Net List Output** function. **Connected Constructive Pins** triggers the output of constructive part pins which are connected to nets. This allows for constructive parts and connections which have been added in the **Layout Editor** to be included in netlist outputs. When re-importing such (modified) net lists, the net names are preserved (which is not the case when using the **Back Netlist** function).

The **Net List Output** function has been optimized to increase the performance of sorted outputs for layouts with large numbers of nets and/or parts.

Layer Stackup (BAE HighEnd)

The **Export** button in **Settings / Rule Attachment / Layer Stackup** for exporting the layer stack configuration to a generic text file has been renamed to **Dump** umbenannt. The **Export** and **Import** buttons for exporting and importing the layer stack configuration to and/or from **.csv** files have been added to the **Layer Stackup** dialog. This allows for layer stack configurations to be exchanged between different projects and/or for the modification of layer stack configurations with external tools.

4.5 Graphic, Copper Areas

Draw Assistant

The **P** button for using the selected pick preference layer as default layer for creating new and picking existing polygons has been added to the **Utilities / Draw Assistant** function.

Options for specifying keepout area sub-types for copper fill and autorouting have been added to the **Keep Out Area** selection menu.

Polygon Corner Change

With DRC clearance display activated, some functions from the **Areas / Other Functions / Polygon Corner Cut** submenu aborted without polygon modification and issued an error messages. This problem has been fixed.

Polygon Combination

The functions from the **Areas / Other Functions / Polygon Combination** submenu sometimes failed to process polygon combinations with adjacent corner points and/or adjacent segments correctly. This problem has been fixed.

The **Process Mode** submenu function for activating different polygon processing modes only set the **Save Defaults** mode correctly. This problem has been fixed. The **OK** dialog button sets the selected processing mode for the current session.

DXF Import

The **DXFIDFFILE_LAY** parameter for setting a default file name for the **AutoCAD/DXF Input** function has been added to **bae.ini**. This feature can be used to, e.g., establish a specific directory for constructive data exchange with a mechanic CAD system, since the file browser start directory is derived from the DXF input file name.

4.6 Text, Drill

Multiline Texts

The `MTEXTORIG_STDbae.ini` parameter for re-configuring multiline text pick points from bottom left to top left was not correctly adhered to by the `p` key dialog for editing multiline texts. This problem has been fixed.

DRC (BAE HighEnd)

The design rule check in **BAE HighEnd** calculated wrong surrounding DRC rectangles for mirrored, vertically centered texts on signal layers. This problem has been fixed.

Drill Legend

The `DRLHTSIZE` parameter for specifying a drill text size different from the legend text size for drill plans created with the `File / Import/Export / Drill Data Output / Drill Plan/Legend` function has been added to `bae.ini`.

A drill class selection has been added prior to the `File / Import/Export / Drill Data Output / Drill Plan/Legend` to allow for the generation of drill class specific drill plans/legends.

External Document References

The `Text, Drill / Other Functions / Link to External File` function has been modified to support file links with space characters in the file and/or directory name.

4.7 Group Functions

Clipboard (Windows)

The **Copy Coordinates** function for copying the coordinates of the group-selected traces and areas as text to the **Windows** clipboard has been added to the **Edit** menu. The coordinate list text format is identical to the format of the **Ctrl-V** function. The **Copy Coordinates** function can be used to transfer a logo consisting of different areas via the clipboard to a different layout hierarchy level or even into the **Schematic Editor** (separate copy/paste steps for lines and areas may be required).

The **Copy Coordinates** can be called through the **Shift-Ctrl-C** key.

Group Report

The **Edit / Other Functions / Group Report** function for generating a selected group element report with selection counts, listings of group-selected macro references and texts and a layer-specific group element listing has been added.

The **Edit / Other Functions / Group Report** function can be used together with the **Ctrl-A** key for group-selecting all elements to display complete layout element reports.

Group Selection

The **Texts** option for selecting and/or deselecting text elements through text string specifications has been added to the **Edit / Other Functions / Select/Deselect / Name** functions.

The name selection boxes of the **Edit / Other Functions / Select/Deselect / Name** function have been modified to list only names of elements which don't have the desired group status. This feature doubles as query function for selected and/or unselected group elements.

The **Parts from File** option for group-selecting parts from a selectable text file with a part name in each line has been added to the **Edit / Other Functions / Select / Name** function.

The **Last Changes** function has been added to the **Edit / Other Functions / Select** and **Deselect** functions. **Last Changes** provides a dialog with a group selection menu for the elements which were created and/or modified in the previous, up to maximum undo count, operations. This dialog supports multiple selections to allow for groups of changes to be selected. Only element operations which were not affected by subsequent operations are selectable.

Group Element Rotation

Functions for left and right rotation and for setting the rotation angle of group selected parts and texts have been assigned to the **L** and **R** keys if those keys are pressed while no other function is active.

DXF Export

The **Group Elements Only** option for restricting DXF outputs to group-selected elements has been added to the **AutoCAD/DXF Output** function.

The **Item Browser** option for restricting DXF outputs to the element types which are currently visible through **Item Browser** settings has been added to the **AutoCAD/DXF Output** function.

Please note that the new option/parameter settings must be included in **AutoCAD/DXF Output** call sequences of existing menu, key, favorites and macro definitions to preserve fully automated **AutoCAD/DXF Output** processes.

4.8 Automatic Copper Fill

Automatic Copper Fill Area Deletion

A fill status marker is now added to fill area outlines when copper fill structures are added to or removed from such fill areas. The copper fill functions have been modified to remove existing copper fill structures from fill areas before (re-)filling such fill areas to prevent the system from generating overlapping and/or redundant copper fill structures.

Copper Fill Area Types

The **Change Fill Type Only** fill area option for configuring selectable fill areas for filling or hatching (regardless of the subsequently called fill function type) has been added to the **Areas / Other Functions / Set Polygon Type/Net** function and to the **p** key dialog. This allows for the **Fill all areas** function to be used to fill and/or hatch different fill areas in one go.

Copper Fill Area Hatching

The line and grid hatching options of the copper fill functions sometimes omitted fill area outline trace segments. This problem has been fixed.

Net Assignment

The **Add Copper Fill Area** function and the functions for changing fill area nets (e.g., from the **p** key dialog) have been modified to automatically derive a net name from the net's first part pin if an unnamed net (e...) is selected. This eliminates the fill net assignment disambiguities which were previously caused by net re-numbering due to net list changes (users frequently used the **Pin Net** options from the **Areas / Other Functions / Set Polygon Type/Net** as a workaround to fix such disambiguities).

The **Areas / Other Functions / Fill Area List** function has been modified to list fill area net names with the part and pin names from which the net names are derived. This allows for more extensive fill net assignment checks.

DRC Blocks (BAE HighEnd)

The copper fill functions sometimes caused distance violations for nets with DRC block specifications and clearance settings lower than the standard layout DRC clearance setting. This problem has been fixed.

Heat traps with shortened connections were sometimes created for parts and/or vias with specific DRC block assignments and clearance specifications higher than the heat trap isolation distance setting, thus failing to connect pins/vias to copper fill areas. This problem has been fixed.

Heat Traps

The **pad_htwidth** predicate for setting element-specific heat trap connection widths has been added to the layout rule system. The **pad_htwidth** predicate can be assigned to pads on padstack level, pins on part level and parts on layout level.

Vias

Vias with fill net assignments which were not yet connected to the fill net were sometimes isolated with the standard layout isolation distance instead of the fill net clearance distance. This problem has been fixed.

Keepout Areas

Keepout areas on signal layers for which either the **Full Check** or the **Cross** option is set were not considered by the automatic copper fill functions. This problem has been fixed.

Context Functions

The **Fill single area** and **Clear single area** functions have been added to the context menus for fill areas.

The **Add Copper Fill Area** function has been added to the context menu for empty areas.

5 CAM Processor

5.1 General

EPS/PDF Output

The **Annotation Output** option of the **EPS/PDF Output** function has been modified to export part PDF layer name settings also with part and pin annotations. This ensures that the visibility of part and pin annotations changes with the visibility of their parts when fading in and/or fading out layers in **Acrobat Reader**.

The **Ignore Element PDF Layers** control for suppressing the output of element specific PDF layers for structures on certain BAE layers has been added to the layer configurations for batch output definitions with PDF layer support.

The layer parameters for batch definitions have been extended to allow for element classes being selected for EPS/PDF output according to their visibility settings in the **Layout Editor** **Item Browser**. This feature can be used to output documentary texts and graphics in different colours through different layer definitions, place fanout trace connections on part level on a special PDF layer, output keepout and fill area outlines as displayed on screen, etc.

CAM Processor Startup

The **CAM Processor** has been modified to support startup with a command line such as

```
> bae cam DDB-filename layoutname ULC:commandsequence ↵
```

or

```
> bae cam DDB-filename layoutname ULC_QUIT:commandsequence ↵
```

for loading a specific layout from a DDB file into the **CAM Processor** and executing the specified command sequence. This allows for the implementation of batches such as fully automated direct manufacturing data outputs. **ULC_QUIT** causes the **CAM Processor** to close down after executing the command sequence.

An asterisk character (*) can be used for the layout name in the above **Layout Editor** startup commands to refer to the default layout element name configured through the BAE setup.

CAM-Batch Output

The **Variant Output Loop** option for iterating all variants of a project has been added to the **CAM-Batch Output** function. Please take care to specify variant-specific output file names when using the **Variant Output Loop** option to prevent variant output files from being overwritten.

The **Total Report** and **Report Variant** options have been added to the **Report Output** batch step to allow for variant-specific reports for each single variant and/or total reports for all variants to be generated with **Variant Output Loop**.

Some bitmap-oriented batch steps such as **Report Layer Coverage** and **Generic/Windows Printer Output** used to reset the highlight and the reported number of overdraw errors from previous Gerber and HPGL outputs. This problem has been fixed to ensure that the overdraw errors from all batch steps are listed in the final batch report.

5.2 Control Plot

Filled Board Outline

The `Border with Cutout` option for displaying the filled board outline in black and the board outline itself in white has been added to the `Monochrome Bitmap Output` facility of the `CAM-Batch Output` function. This is useful for displaying PCBs with cutout milling contours. Note that closed polygons are still filled in white when using the `Border with Cutout` option as this reflects the real milling result.

The layer definition facility of the `EPS/PDF Output` function has been modified to support the new `Fill Board Outline` option for board outlines and the new `Fill Closed Lines` option for documentary layers to allow for the configuration of filled PCB displays with milling cutouts as well.

Generic Output (Windows)

The `Generic/Windows Printer Output` function has been modified to allow for the specification of a printer name after the layer selection in macros and key programming sequences. The name `standard` can be used to select the `Windows` default printer.

The `Generic Output Drawing Mode` option for controlling the color management for multi-layer plots with `Generic Output` has been added to the `Plot Output / Settings` dialog. `Opaque Color` is the default setting for opaque coloring according to the layer selection sequence. The `Merged Colors` mode creates a bitmap using mixed colors on a black background and then switches white and black before performing the actual print output. This allows for mixed color outputs without black backgrounds thus saving toner and/or ink. Please note that the `Merged Colors` mode might require significant main memory and computing resources due to the necessary bitmap operations.

The `#214` call has been implemented to allow for `Generic Output Drawing Mode` settings in `Generic Output` macro and key programming sequences.

The `CAM-Batch Output` function has been modified to support `Generic/Windows Printer Output` batch steps.

6 CAM View

6.1 Data Import

Input Offset

The dialog of the `File / Batch / Load` function accepted only integer values for input offset specifications. This problem has been fixed.

7 Utilities

7.1 BSETUP

Documentary Layers

The `DOCUMENTU` command has been added to allow for the assignment of documentary layers to the top level of the layer selection menus.

7.2 LISTDDB

Element Size

The output of the `LISTDDB` utility program has been extended to include the spatial requirements for storing elements in the DDB file. This information is useful when using `LISTDDB` outputs to compare the content of different DDB files.

7.3 LOGLIB

Pin Command

The

```
pins none;
```

command for suppressing automatic 1:1 assignments of symbol to layout pins for missing `pin` commands has been added to the logical library definitions. This allows for the definition of general purpose `mainpart` symbols without pins.

8 Bartels User Language

8.1 General

This section describes general changes to the **User Language** specification. See [Bartels User Language Programmer's Guide - Chapter 2](#) for a detailed description of the **User Language** specification.

Internal User Language Version

The internal **User Language** version has been changed. **User Language** programs compiled under BAE versions prior to BAE V7.8 won't execute in the **User Language Interpreter** environment of the new **Bartels AutoEngineer** version (error message `User Language program version incompatible!`) and must be recompiled under the new **BAE** version to regain compatibility.

User Language programs compiled under **BAE V7.8** and later can execute in the **User Language Interpreter** environment of the new **Bartels AutoEngineer** version without being re-compiled. However, it is recommended to re-compile these programs as well to ensure that the latest **User Language** features are enabled.

8.2 System Functions

This section lists new and changed **User Language** system functions. See [Bartels User Language Programmer's Guide - Appendix C](#) for a detailed description of all system functions.

New System Functions

The following **User Language** system functions have been implemented:

IP	System Function	Short Description
STD	bae_msgprogressrep	Activate/update BAE progress display
STD	bae_msgprogressterm	Terminate BAE progress display
STD	strgetconffilename	Get environment variable expanded configuration file name
CAP	cap_getscbustapidx	Get currently scanned SCM bus tap
SCM	scm_checkbustapplot	Get SCM bus tap plot status
LAY	lay_getscclass	Get currently scanned layout element class
LAY	lay_setfigcache	Fill layout figure list access cache

Changed System Functions

The maximum number of menu entries for the [bae_askmenu](#) function has been increased from 30 to 50.

The file access mode parameter of the [fopen](#) function has been modified support file autoclose modes (for **BAE Demo** write access checks).

The [remove](#) function has been modified to allow not only for the deletion of files but also for the deletion of empty directories.

The [scandirnames](#) function has been modified to refresh its internal file name cache not only when the directory name or the file name extension is changed but also if a new scan of the previous directory is requested through an empty file name specification. This ensures that directory content changes are registered between consecutive scan requests for the same directory.

The [dirscan](#) kept scanned directories locked, thus blocking the deletion of scanned directories until the calling BAE module was terminated. This problem has been fixed.

Support for additional parameter queries/settings has been added to the [bae_getdblpar](#), [bae_getintpar](#), [bae_setintpar](#), [bae_getstrpar](#), [bae_setstrpar](#), [scm_getintpar](#), [scm_setintpar](#), [scm_getstrpar](#), [ged_getintpar](#), [ged_setintpar](#), [ar_getintpar](#), [ar_setintpar](#), [cam_getintpar](#) and [cam_setintpar](#) functions.

The pick element type parameter of the [scm_pickelem](#) function has been extended to support the pick of symbol attributes.

8.3 BAE User Language Programs

BAE installs more than 220 pre-compiled **User Language** programs to the `ulcprog.vdb` file of the BAE programs directory. Additionally, the **User Language source** files (more than 10 Mbytes; some 330,000 lines) are installed to a special directory (`baeu1c`). See [Bartels User Language Programmer's Guide - Chapter 4](#) for a complete listing and short descriptions of the BAE User Language programs.

User Language Include Files

The **User Language** include files have been revised and extended by a series of new definitions and functions.

New User Language Programs

The following **User Language** programs have been implemented:

IP	Program Name	Short Description
STD	CMDCALL	Execute Command Call Sequence
GED	MT_ROUT	Mikami-Tabuchi Router

Changed User Language Programs

The **User Language** programs already delivered with the previous BAE Version have been revised and extended by many new features and functions.