



Bartels AutoEngineer®

Version 6.4

Release Notes

This documentation contains information about the new features introduced with **Bartels AutoEngineer Version 6.4**. Forward compatibility from earlier versions to **Bartels AutoEngineer Version 6.4** is ensured, but not backward compatibility.

Bartels AutoEngineer Version 6.4 Release Notes

Published by: Bartels System GmbH, Erding

Last printing: August 2004

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 © 1986-2004 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	6
2	Schematic Editor	7
2.1	General	7
2.2	Symbols, Labels	8
2.3	Connections	10
2.4	Graphics.....	11
2.5	Text, Attributes	11
2.6	Group Functions.....	12
2.7	Plot Output	12
2.8	Hierarchical Circuit Design	12
3	Packager	13
3.1	Test Point Generation	13
4	Layout Editor	14
4.1	General	14
4.2	Display, Design Rule Check.....	15
4.3	Parts, Placement.....	16
4.4	Traces, Routing.....	17
4.5	Graphic, Copper Areas.....	18
4.6	Text, Drill	19
4.7	Group Functions.....	19
4.8	Automatic Copper Fill	20
5	Autorouter	21
5.1	Autorouter Algorithms.....	21
6	CAM Processor	22
6.1	General	22
6.2	Control Plot	22
6.3	Gerber Photo Plot.....	22
7	CAM View	23
7.1	General	23
7.2	Data Import	23
8	Utilities	24
8.1	BSETUP.....	24
8.2	LOGLIB	24
9	Bartels User Language	25
9.1	General	25
9.2	User Language Interpreter	25
9.3	Index Variable Types.....	25
9.4	System Functions.....	26
9.5	BAE User Language Programs	27

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. Parameter Settings

The `bae.ini` file has been changed to allow for the configuration of parameter lists. Previously, this could only be done in **User Language** source code files, which meant that these had to be edited and/or recompiled with every BAE update. With parameter lists configured in `bae.ini` it is not necessary anymore to edit and/or recompile any such settings such as SCM plan header attributes, part list output attribute lists, fixed tool tables for drilling data output, etc.

1.2 User Interface and General Functions

Mouse Button Function Assignment

The `Pulldown RMB Context` option has been added to the `Menu type` parameter in the `Setup` dialog of the BAE main menu, and the `Pulldown` option has been renamed to `Pulldown LMB Context`. `Pulldown LMB Context` activates the BAE pulldown menu interface with context menus assigned to the left mouse button and function repeat assigned to the right mouse button. `Pulldown RMB Context` activates the BAE pulldown menu interface with context menus assigned to the right mouse button and function repeat assigned to the left mouse button, a setup with which Windows users might be more familiar.

The input format of the `BSETUP` utility program has been extended accordingly. I.e., the `PULLDOWN_RMB_CONTEXT` option has been added to the `WINMENU MODE` command.

Grid Placement

`Shift`/arrow key combinations can be used during interactive placement to move the placement object to the next grid position in arrow direction. The `Enter` key finishes the placement operation at the current position.

Pressing the `Enter` key whilst holding down the `Shift` key is equivalent to left mouse button clicks (as, e.g., applied for setting polygon corner points without exiting the polygon definition function). Pressing the `Enter` key whilst holding down the `Ctrl` key triggers a `Center/Pan Window` to the mouse position and moves the mouse cursor to the center of the window.

Design View Windows

Design view window configurations with references to other than the current element are now being saved with the currently loaded element. The reference element and file names are displayed when reloading such design view windows; the referenced element needs to be loaded to update the design view window.

Key Programming

The `Key Programming` function has been added to the `Utilities` menus of the BAE modules.

Menu Programming

The `Menu Extension` function for adding menus and/or menu functions to the current user interface has been added to the `Utilities` menus of the BAE modules and a button for activating this feature has been added to the dialog which can be activated through shortcut key `F5`. `Menu Extension` can be used to make frequently used submenu functions more accessible and to integrate user-defined macros and **User Language** programs into the menu.

Favorite Menu Configurations

New buttons for saving and loading different favorite menu configurations have been added to the favorites menu definition dialog. This allows for the definition and/or activation of different favorite menus for different tasks such as library element definition, schematic capture or layout placement.

Command Repetition

Automatic command repetition through right mouse button clicks has been deactivated for commands for which repeated calls don't have any effect or don't make sense such as `Save` or `Settings`.

Undo Buffer

The Setup dialog from the BAE main menu has been changed to allow for the number of Undo steps to be set in between 20 and 100. Different Undo buffer sizes can be specified for the **Schematic Editor** and the layout system.

Data Security

To prevent from irrecoverable loss of data due to unintentional use of file element access functions, the functions for deleting and copying elements and for saving elements and/or groups under different names have been changed to automatically create backup copies of the deleted and/or overwritten `.ddb` file elements in a corresponding `.bak` file.

Window Positions

BAE window configurations are now saved separately for each BAE module. I.e., different window positions and dimensions (possibly on different screens if using dual/multi-monitor systems) are restored when switching between and/or restarting different BAE modules such as the **Schematic Editor** and the **Layout Editor**.

Element Name Selection

The `Pattern` for restricting the list of menu entries to a name pattern such as `dil*` or `*16` has been added to the DDB element selection dialogs.

DDB Element Copy

The source file selection dialog of the `Copy Elements` function from the `Library Utilities` submenu of the **Schematic Editor** and **Layout Editor File** menu has been changed to allow for the selection of any (non-DDB) file names. I.e., it is now possible to copy DDB file elements from, e.g., `.bak` files without having to leave the BAE user interface. A dialog with Windows standard features for selecting multiple entries has been provided for the element selection. This dialog also provides a button for the selection and/or deselection of elements through element name patterns.

Project File Name Display

The `TITLEMODE_STD` entry for activating a project file name display (without file name extension and directory path) in the BAE title bar has been added to the `bae.ini` file. This simplifies the selection of BAE sessions through the Windows task list/bar if multiple project files are open at the same time.

The name of the currently processed project file is now displayed in the title bar of the BAE sidemenu configurations.

1.3 Symbol and Part Libraries

ROUTE - Router Control, Rule System

The tag symbols `tag_net_netfill` (for assigning layout copper fill parameters to SCM signals), `tag_netarea_*` (for assigning net attributes to pin nets withing a net range area), `tag_netpin_*` (for assigning net attributes to pin nets), `tag_net*_routdisable` (for `$routdis` net attribute assignments), `tag_net*_testdisable` (for `$notest` net attribute assignments), `tag_net*_viapadstack` (for `$viastk` net attribute assignments), and `tag_net*_routlayers` (for `$layers` net attribute assignments in **BAE HighEnd**) have been added to the **ROUTE** symbol library.

STDSYM - Standard/Template Symbols, Labels

SCM sheet templates (A0, A1, A2, A3, A4, letter and tabloid landscape and portrait formats) and the `planhead` SCM plan header symbol have been added to the **STDSYM** library.

2 Schematic Editor

2.1 General

Pick Point Display

The **Pick Point Display** entry with the **No Pick Point Display** and **Pick Point Display** options has been added to the **Settings** dialog of the **View** menu. **Pick Point Display** activates a pick and corner point display with small circles being used to indicate pick and corner points. Indicators for text pick points, connection corner points and polygon corner points are displayed in the corresponding element color. Indicators for symbol origins and polygon arc center points are displayed using the currently selected color for origins. **No Pick Point Display** deactivates the pick and corner point display.

Dynamic Element Boundaries

The **Element Boundary** entry with the options **fixed** and **dynamic** has been added to the **Settings** dialog from the **Settings** menu. The default option **fixed** does not allow for any objects to be placed outside the element boundaries. The new **dynamic** option allows for the placement of objects outside the element boundaries, in which case the element boundaries are automatically adjusted.

Default Parameter Settings

The **DEFBUSDISP_SCM** (bus display mode), **DEFSYMMUM_SCM** (symbol numbering mode), **DEFSYMTMM_SCM** (symbol position mode) and **DEFSIGROUT_SCM** (signal router setup) default parameter entries for new SCM sheets have been added to the **bae.ini** file.

Automatic SCM Sheet Frame Insertion

A new feature for automatically offsetting the plan origin and loading a frame graphic has been added to the function for creating new SCM sheets. For landscape formats, the plot rotation is automatically set to 90 degrees. Defaults for the origin offset, the SCM library and the frame graphic/group for standard sheet sizes can be configured through new **bae.ini** entries.

New SCM plan frame groups have been added to the **STDSYM** default/standard SCM library. These frame groups are using the **planhead** header symbol with attribute definitions suitable for the **Set Frame Attributes** function. The **header** group name is assigned to the SCM plan frame elements to allow for quick selection.

Keyboard Input Coordinate Confirmation

The enter/return key can now be used alternatively to the left mouse button to confirm and/or enter the current mouse position. When editing connections and/or graphic lines, the current input coordinate is added to the input point list and the connection and/or line input function is terminated.

Toolbar Drawing Functions

New symbols (line, dashed line, area, and **T** for texts) for quick access to the **Add Graphic Line**, **Add Dot Line**, **Add Graphic Area** and **Add Text** (standard or comment text) have been added to the **Schematic Editor** toolbar.

Glue Elements

The **Glue** function with the **Glue Symbol(s)**, **Unglue Symbol(s)**, **Glue Group Elements**, **Unglue Group Elements**, **Select Glued** and **Deselect Glued** options has been added to the **Rule Attachment** function from the **Settings** menu. This allows for the placement of elements to be permanently fixed ("glued"). The positioning of glued elements cannot be changed, nor can glued elements be deleted. Only the re-positioning of name and attribute texts for glued parts with rotation-independent text visibility is allowed. Texts on glued symbols with rotation-dependent text visibility are automatically glued to prevent from unintentional editing.

Global Net Highlight (BAE HighEnd)

The **Highlight Net** function from the **BAE HighEnd View** menu has been changed to support cross-probing (project layout net highlight synchronization) for unnamed nets. With this feature, the **Highlight Pin Net** function from the **Other Functions** menu became redundant and was removed.

2.2 Symbols, Labels

Symbol Name Extension

The **Append Suffix** and **Delete Suffix** options for automatically appending and deleting part name suffices have been added to the **Renumber Parts** function from the **Other Functions** submenu of the **Symbols** menu.

Automated Pin Placement

The **Place Pin List** function from the **Symbol Edit Tools** submenu of the **Symbols** menu has been renamed to **Place Pin List/Row** and new buttons for automated pin row placement have been added. Automatic pin placement has been extended to support not only numeric pin numbering but also pin naming based on alphanumeric index name patterns. The **Place Pin Texts** function has been extended accordingly.

Attribute Locking

A new button for optionally locking attribute value assignments has been added to the **Attribute Default Values** dialog of the **Set Attributes** function from the **Other Functions** submenu of the **Symbols** menu. The **Assign Value(s)** function still displays locked attributes but doesn't allow to change them anymore. Locked attribute value assignments can be used as an alternative to fixed attribute value assignments through the logical library `newattr` command. Unlike fixed attributes, locked attribute values can be included with SCM displays and/or printouts.

New procedures for locking and/or unlocking attribute value assignments have been added to the **Symbol Edit Batch** function.

Net Name Selection

The system now automatically performs a **Project Scan** when loading a schematic sheet to ensure that the project net name selection menu for the **Add Label** function is always up to date.

Cross Reference List

A column for displaying symbol attribute values has been added to the symbol list output of the **SCM Cross Reference** function from the **Other Functions** submenu of the **Symbols** menu.

The `$pageref` system attribute has been added for SCM labels. On SCM sheet level, the `$pageref` attribute value is automatically substituted with the list of the SCM sheet names on which the net is used.

Edit Symbol Logic

The **Edit Symbol Logic** function has been changed to prompt for the layout part macro name for a default logical library definition if applied on a symbol without logical library definition. The layout part macro can be selected from a dialog which lists the layout part macros from the project file and the default layout library. The pins of the selected layout part macro are stored to the logical library definition for easier logical library definition editing.

The **Edit Symbol Logic** function automatically loads the first layout part macro from the logical library definition to the project-specific **Layout Editor** window if used in **BAE HighEnd** and if a project-specific **Layout Editor** window is active.

Symbol Browser

< and **>** buttons for switching to the previous or next symbol of the currently browsed library have been added to the symbol display dialog of the **Symbol Browse** function from the **Other Functions** submenu of the **Symbols** menu.

Symbol Database

The **Symbol Pool** button for activating facilities for managing a repository of parts for subsequent placement has been added to the **Symbol Database** function. The **Next Symbol/Label** has been changed to place a part from this symbol pool which best matches the previously placed part. Parts from the symbol pool are removed once they are placed. The symbol pool is saved with the DDB project file, i.e., it can be used throughout different **Schematic Editor** sessions. The symbol pool also supports part definitions with different subsymbol types. E.g., are added to the symbol pool if two a selection of two `1m324` parts adds eight opamps and two power supply symbols to the symbol pool. With this feature, the **Schematic Editor** automatically provides information about free and/or unused gates.

The **Doc** for loading the documentation for the currently selected symbol database entry has been added to the lower selection level of the **Symbol Database** function.

The **Layout Part** button for automatically loading a **Layout Editor** window for viewing the layout part definition for the currently selected SCM symbol has been added to the **Symbol Database** function in **BAE HighEnd**.

Symbol Routing

A significantly improved cost-based symbol routing algorithm for re-routing connections to moved symbols and/or groups has been implemented. The new **Signal Router Mode** entry from the **Settings** dialog of the **Settings** menu can be used to select the signal routing algorithm. **Full Evaluation** is the default option for activating the more accurate (and more time-consuming) new signal routing algorithm. The **Quick Shot** option activates the less accurate, yet quicker signal routing algorithm. **Quick Shot** is intended for less powerful systems where the new signal router might cause delays when dealing with long distance symbol movements.

The system issues a warning message indicating the number of unrouted connections if it fails to re-route all connections,

Symbol Rule Transfer

The toolbar and context menu functions for copying symbols and/or attributes have been changed to include the transfer of symbol rules such text class masks or plot visibility.

Layout Part Display (BAE HighEnd)

The **DISPPLNAME_SCM** parameter has been added to the **bae.ini** configuration file for **BAE HighEnd**. This parameter setting causes **BAE HighEnd** to activate a layout part symbol display in a new **Layout Editor** window for any **\$plname** attribute selection through the **Browse** button.

2.3 Connections

Pin Connections

The **Connectivity** function has been added to the **Rule Attachment** submenu of the **Settings** menu. **Standard** is the default option, allowing for connections only to be created by adjoining connection end points. The **Full Pin Check** option allows for connections to be created by placing pins onto each other (e.g., to connect pins in a series circuit, to place labels directly onto part pins, etc.) or by drawing connections segments over more than two pins (e.g., to connect pins in a parallel circuit). With the activation of the **Full Pin Check** mode, the system issues a warning about the dangers of unintentionally connecting pins when switching display zoom factors whilst drawing connections. Changes to the **Connectivity** setting are automatically applied to all SCM plans of the current project by loading and saving all project plans with the new connectivity.

Connection Marker Display

A T-connection marker is now displayed at pins which are connected through 90-degree connection segments.

Bus Tap Name Selection

The bus tap functions for named busses have been changed to include the bus taps from all SCM sheets of the project rather than only the current sheet's bus taps in the bus tap selection menu.

Bus Tap Processing

Bus tap support has been added to the context function menus to be activated by left mouse button element clicks.

The group highlight, group re-routing and plot visibility control functions sometimes processed unselected bus taps. This problem has been fixed.

Bus Tap Macro

The **Select Bustap Macro** option for selecting a (non-default) label macro for subsequent bus tap placements has been added to the **Settings** dialog from the **Settings** menu. The **Net Attribute Label** tag mode for bus tap labels transfers fixed attributes to the net connected to the bus tap.

The context menu available through right mouse button clicks during bus tap placement operations provides the **Change Macro** function for changing the label macro of the currently placed bus tap.

The **Bus Tap Label** option for changing the label macros of all currently group-selected bus taps has been added to the **Group Macro** function from the **Edit** menu.

Net Attribute Assignments

The **Net Pin Tag** and **Net Area Tag** options have been added to the **Tag Pin Function** from the context menu available through right mouse button clicks during tag symbol pin placement operations.

Net Pin Tag can be used for net attribute assignments. Unlike **Net Tag**, **Net Pin Tag** targets net symbol pins which allows for the assignment of attributes to unnamed nets.

Net Area Tag can be used to assign net attributes to all nets with pins inside a specific net range area. Net range areas are defined by assigning a net range name to an area created with the **Add Graphic Area** function using the **Net Area** function from the **Rule Attachment** submenu of the **Settings** menu. Net range areas are drawn with dashed outlines and include the net range name display at the outline start point.

New tag symbols, `tag_netpin_*` (tag symbols for assigning net attributes to pin nets) and `tag_netarea_*` (tag symbols for assigning net attributes to pin nets withing a net range area), have been added to the **ROUTE** symbol library.

Copper Fill Parameter Assignments

A new tag symbol named `tag_net_netfill` for assigning copper fill parameters to SCM signals has been added to the **ROUTE** library. Copper fill parameters assigned through `tag_net_netfill` have priority over standard copper fill parameters, but are overridden by layout copper fill area parameter assignments.

2.4 Graphics

Keyboard Coordinate Input

The **Loop** option has been added to the **Jump relative** and **Jump absolute** context menu functions for creating and/or editing polygons and lines. **Loop** activates coordinate input prompts until **Abort** is selected. This allows for the keyboard input of a series of coordinate points without having to switch between mouse and keyboard.

Polygon Corner Processing

The **Direction** and **Angle Direction** functions for changing the edit direction or for switching the angle-locked segment have been added to the context menu for creating and editing areas and lines.

Arcs with Cut Segments

A redundant **Invalid polygon!** error was issued in special cases when drawing closed polygons consisting of one arc segment and a line segment at the end. This problem has been fixed.

DXF Line Import

A feature for automatically joining subsequent lines and polygons with identical end and start point coordinates has been added to the **AutoCAD/DXF Input** function from the **Import/Export** submenu of the **File** menu. Any DXF line width settings are now also converted accordingly.

2.5 Text, Attributes

Load Text Sheet

The **Load Text Sheet** for loading project SCM sheets by clicking texts has been added to the **Other Functions** submenu of the **Text** menu and to the text context menus. **Load Text Sheet** checks the words of the selected text for an existing project file SCM sheet name and loads the first matching SCM sheet. This allows for the definition of symbolic (hyper-)links between different SCM sheets or for the definition of a central SCM plan index sheet.

Symbol Text Class Mask

The **Text Class Group Assignment** function from the **Other Functions** submenu of the **Text** menu has been changed to activate a submenu consisting of the existing/previous **Text Assignment** function and the new **Symbol Mask** function for setting the text class mask for all currently group-selected symbols.

Default Attribute Values

Assigning **\$** as default attribute value for **\$rpname** attributes causes the system to assign the symbol name to the **\$rpname** attribute when adding a symbol on the schematic sheet.

Project File Name Attributes

The **\$pltfbname** (File Base Name) and **\$pltfbsname** (File Base Short Name) system attributes for displaying the current element's project file name have been added. **\$pltfbname** displays the project file name without the **.ddb** extension. **\$pltfbsname** displays the project file name without the **.ddb** extension and without the directory path.

Mirrored Symbol Text Positioning

The **Disolve Rotations** and **Combine Rotations** from the **Rule Attachment** submenu of the **Settings** menu have been added to support alternative text positioning for unmirrored and mirrored symbols.

2.6 Group Functions

Clipboard Data Reference Point

The **Cut** and **Copy**, when activated through **Ctrl-x** and **Ctrl-c** keys, are now using the current mouse position as reference point for the clipboard data. This allows for parts of the schematic plan/element to be copied from one (e.g., right top) corner to another (e.g., left bottom) corner of the element without having to modify the element boundaries for this placement operation. **Cut** and **Copy** still use the plan origin as clipboard reference point if called from the **Edit** menu.

Clipboard Data Name Preservation

SCM sheets are now saved without the part names of symbols which have previously been transferred to the clipboard using **Cut** functions. This allows for the parts/symbols from the clipboard with their original part names to be pasted onto another sheet of the same project without causing any part name collisions.

Name/Attribute Positions

The **Transfer Name/Attribute Positions** function for transferring name and attribute text placement modifications from a selectable symbol to all group-selected symbols with the same macro type has been added to the **Other Functions** submenu of the **Edit** menu. This function comes in handy if specific part name and attribute text positioning needs to be applied to a group of parts such as a series of block capacitors.

Label Selections

On SCM plan level, the **Symbols** and **Labels** modes have been added to the **Names** option of the **Select** and **Deselect** functions from the **Other Functions** submenu of the **Edit** menu. **Labels** allows for the selection of net groups through net name patterns for subsequent label macro replacement.

2.7 Plot Output

Plot Parameter Memory

SCM sheet specific plot rotation parameter settings are now saved with each SCM sheet. This simplifies the setup of batches for plotting all and/or different landscape/portrait SCM sheets of a project.

2.8 Hierarchical Circuit Design

Load Block Plan

The **Load Block Symbol** and **Load Block Plan** options for loading either the hierarchy block symbol or the hierarchy block plan have been added to the **Load Macro** context menu function for block plan symbols.

Block Symbol Creation

The **Load Block Sheet** function from the **Other Functions** submenu of the **Symbols** menu has been replaced by the **Block Sheet** submenu with the **Load Block Sheet** and **Create Block Symbol** functions. **Load Block Sheet** is the existing function for loading hierarchy block plans. **Create Block Symbol** is a new function for automatically generating a symbol for the currently loaded hierarchical schematic block sheet. The block symbol pins are placed on the edge of the block symbol together with name texts according to the orientation of the module ports, and a block name text is placed in the center of the block symbol. Additionally, a logical library template definition for the block symbol is generated and loaded for editing.

3 Packager

3.1 Test Point Generation

Test Point Generation

The `Test Point Name Prefix` and `Test Points Logical Library` functions for automatically generating net test points has been added to the `Settings` menu. `Test Points Logical Library` is used to specify the name of a logical library definition for automatically generated test points. The default setting for `Test Points Logical Library` is an empty string which deactivates automatic test point generation. `Test Point Name Prefix` is used to specify a name prefix for automatically generated test points. The netlist "part" names for the test points are created by appending the net name (of named nets) or a dot and the part and pin name of the first net pin (for unnamed nets) to the test point name prefix. The parameters for generating test points are saved with the project, i.e., they don't have to be re-entered in subsequent **Packager** runs.

`$notest` net attribute assignments can be used to deactivate automatic test point generation for selected nets. `tag_net*_testdisable` tag symbols for `$notest` net attribute assignments have been added to the `ROUTE` symbol library.

4 Layout Editor

4.1 General

Dynamic Element Boundaries

The **Element Boundary** entry with the options **fixed** and **dynamic** has been added to the **Settings** dialog from the **Settings** menu. The default option **fixed** does not allow for any objects to be placed outside the element boundaries. The new **dynamic** option allows for the placement of objects outside the element boundaries, in which case the element boundaries are automatically adjusted.

Glue Elements

The **Glue** function with the **Glue Part(s)**, **Unglue Part(s)**, **Glue Group Elements**, **Unglue Group Elements**, **Select Glued** and **Deselect Glued** options has been added to the **Rule Attachment** function from the **Settings** menu. This allows for the placement of elements to be permanently fixed ("glued"). The positioning of glued elements cannot be changed, nor can glued elements be deleted. Only name and attribute text re-positioning is allowed for glued parts. The **Autorouter** treats glued traces and vias like fixed traces and vias. This feature can be used to protect parts such as specifically placed connectors from unintentionally being moved with, e.g., group functions.

Net Name Selection

The **Pick Net** button for selecting nets by clicking an element of the desired net has been added to the net name lists of the net name selection dialogs.

Default Parameter Settings

The **DEFTRCDISP_GED** (trace edit display mode), **DEFVIACRNG_GED** (via checking range), **DEFVIA_GED** (default via padstack), **DEFBRDMODE_GED** (automatic board outline generation) and **DEFBRDENL_GED** (element border expansion for automatic board outline generation) default parameter entries for new layouts have been added to the **bae.ini** file.

Keyboard Input Coordinate Confirmation

The enter/return key can now be used alternatively to the left mouse button to confirm and/or enter the current mouse position. When editing traces and/or creating documentary lines, the current input coordinate is added to the input point list and the trace and/or line input function is terminated.

Position Edit

The snap function assigned to the **P** key can now also be used if no other function is currently active, in which case the coordinates of the snapped element and/or element corner are displayed in a dialog for direct editing. This dialog also provides a rotation angle edit field for parts and texts and a text height edit field for texts.

The functions assigned to the **P** key have been changed to consider any element type restrictions configured through the toolbar mouse button.

4.2 Display, Design Rule Check

Pick Point Display

The **Pick Point Display** entry with the **No Pick Point Display** and **Pick Point Display** options has been added to the **Settings** dialog of the **View** menu. **Pick Point Display** activates a pick and corner point display with small circles being used to indicate pick and corner points. Indicators for text pick points and trace and and polygon corner points are displayed in the corresponding element color. Indicators for part origins and trace and polygon arc center points are displayed using the currently selected color for origins. **No Pick Point Display** deactivates the pick and corner point display.

Grid Origin

The **Element Size** submenu from the **Settings** menu has been renamed to **Workspace** and extended by the **Origin Snap to Input Grid** function for adjusting the origin of the currently loaded element to the internal system grid origin by moving the element inclusive of origin and element boundaries accordingly.

Element Query

The **Query Element** function has been changed to display the documentary layer name (as defined in the BAE setup) instead of the documentary layer number when querying documentary layer elements. **Query Element** has been extended to allow for the querying of documentary lines and the board outline.

For parts and keepout areas, **Query Element** displays any maximum height definitions for the height DRC.

For part pins, **Query Element** displays logical part and pin names (SCM symbol and SCM symbol pin names).

Color Palette Buffers

The **TB_CPALROW_LAY** entry for setting layout toolbar color palette buffers has been added to the **bae.ini** file. The color palette buffers are named **1** through **4**. The **s** (Save) button can be used to save the current color palette to one of the four color palette buffers. The **d** (Default) toolbar button is provided for loading the color table named **standard**. The color setups are stored using the **toolbarn** name patten, thus allowing for color configurations to be loaded to different BAE sessions.

Documentary Layer Settings through Toolbar

TB_DLCLn_LAY entries can be added to the **bae.ini** file to add buttons for quick documentary layer selection to the toolbar of the layout system. Documentary layer selection buttons in the toolbar are complemented by symbols (line, area, "T" for text) for quick access to the **Add Document Line**, **Add Document Area** and **Add Text** functions with automatic documentary input layer selection. The **bae.ini** file shipped with the BAE software configures the first documentary layer (Insertion Plan, see entry **TB_DLCL01_LAY = DOCLAYER 1**) for quick toolbar access.

Height DRC

A new feature for optionally specifying a part-specific distance to the PCB surface in addition to the part height has been added to the **Height Specification** option from the **Height DRC** function from the **Other Functions** submenu of the **Parts** menu, thus allowing for the modelling of part bodies with placement space underneath.

4.3 Parts, Placement

Part Renumbering

The **All Prefixes** option has been added to the **Autoname Parts** function from the **Other Functions** submenu of the **Parts** menu. **All Prefixes** renames/renumbers all netlist parts with names matching a (non-numeric) prefix followed by a number. The new part names consist of the selected prefix and a part number with part (re-)numbering starting at 1 for each prefix.

SMD Generation

New options for automatically generating SMD pads and/or padstacks with different mirror display dimensions and SMD/solder mask size offsets have been added to the **Pad Generator** and **Padstack Generator** procedures of the **Macro Generator** function from the **Library Utilities** submenu of the **File** menu. The size offset and pad name extension parameters can be set through new entries in the `bae.ini` file.

Layout Library Edit Batch

The **Macro References Change**, **Layer Assignments**, **Shrink Macros** and **Enlarge Macros** functions from the **Library Utilities** submenu of the **File** menu have been moved to the new **Layout Library Edit Batch** submenu which also supports a series of new features such as setting text sizes, placing insertion pick texts, assigning rules, etc.

Part Browser

< and **>** buttons for switching to the previous or next part of the currently browsed library have been added to the part display dialog of the **Part Browse** function from the **Other Functions** submenu of the **Parts** menu.

Part Data Snap

The **Complete Part Data Snap** function for transferring the placement data of a selectable part to the currently placed part has been added to the context menu to be activated through the **⌘** key during part placement. This allows for the exact placement of parts onto sockets, cooling devices onto parts, etc.

Automated Pin Placement

The **Place Pin Row** function from the **Other Functions** submenu has been renamed to **Place Pin Row/Matrix** and provides new buttons for automatic pin matrix placement (e.g., for BGA footprint definitions) and for placing a parallel second pin row in reverse order (e.g., for DIL or SO footprint definitions).

The **Place Pin Row/Matrix** and **Place Pin List** function have been extended to support arbitrary alphanumeric pin name indexing in addition to numeric pin numbering.

Library Documentation

Light Format 180*120 sheet size has been added to the output format options of the **Library Documentation** function from the **Import/Export** submenu of the **File** menu. This allows for layout library documentation to be generated with **BAE Light** software.

Attribute Position Display

The **Without Line** and **With Line** options for optionally displaying a line from the original part attribute position to the new attribute position has been added to the context menu of the **Move Attribute** function for moving part attributes. This feature can be used for displaying part name lists/legends at the edge of the PCB with lines to the corresponding parts on the board.

Pin/Gate Swap Display

The display of the **Pin/Gate Swap** function has been modified to use inverted circles for the pins of the selected swap gate. This feature ensures that the selected swap gate is highlighted even if it doesn't contain any swap pins.

4.4 Traces, Routing

Segment Move

During the application of the **Move Segment** function from the **Traces** menu, the implications for neighbouring trace segments are now displayed to allow for a more predictable result when using different **Segment Move** modes.

The left mouse button **Move** operation mode has been extended to allow for trace segment movements as well as for trace corner movements.

Trace Segment Width Change

New functions for increasing and reducing trace segment widths during manual routing have been assigned to the **B** (increase width) and **b** (reduce width) keys. The step value for these operations can be set through the **PATHENLARGE_GED** parameter in the **bae.ini** file. Using the width increase operation immediately after a width reduction, and vice versa, restores the trace's original width. A design rule check is activated for trace segment width reduction requests, and trace segment width reductions with the current step value are automatically carried out until the modification is free of DRC errors. This feature can be used together with the **Fill & DRC Edit Display** mode to reduce a trace width temporarily to route through narrow pin channels ("necking").

The **With DRC** and **All** options have been added to the **Change Path Widths** function from the **Other Functions** submenu of the **Edit** menu. **All** changes the trace widths for all group-selected traces. **With DRC** suppresses trace and/or trace segment width changes which would cause minimum distance design rule check violations. I.e., the **With DRC** option can be (repeatedly) used at the end of the design process to maximize trace widths.

Trace Corner Processing

The **Group Arcs**, **Group 45 Degrees Angles** and **Group Eliminate Arcs** options for manipulating all group-selected traces have been added to the **Trace Corner Cut** function from the **Other Functions** submenu of the **Traces** menu.

Parallel Traces and Trace Patterns

The **Continue Pattern** function has been added to the **Parallel Traces** functions from the **Other Functions** submenu of the **Traces** menu. **Continue Pattern** continues the pattern defined by two selectable traces with identical numbers of corner points by creating traces with equidistant corner points. This allows not only for the automated creation of equidistant parallel trace bunches but also for the transfer of trace bunches onto a different grid.

Via Macro Selection

Blind and buried vias for layer changes during manual routing were selected according to the source and destination layer numbers. This could cause the selection of vias with redundant pads for layer setups with nonlinear signal layer arrangements. A feature for considering the number of inside signal layers with pads has been added to the via selection facility to solve this problem.

Via Processing

The **Load Macro** function for loading the padstack macro of a selectable via has been added to the **Via Functions** menu and the via context menus.

The **Place Vias** and **Move Vias** functions from the **Via Functions** have been changed to display the via during placement operations.

Global Net Highlight (BAE HighEnd)

The **Highlight Nets** function from the **BAE HighEnd Highlight Net** submenu has been changed to support cross-probing (project SCM net highlight synchronization) for unnamed nets.

Layer Stackup (BAE HighEnd)

The **BAE HighEnd Layer Stackup** function (see **Rule Attachment** submenu from the **Settings** menu) has been changed to allow for layer comment assignments. For power layers without existing comment, the power layer net name and power plane net names (if any) are automatically assigned.

The **Export** button for exporting the current layer stackup definition with all parameter settings to a text file has been added to the **Layer Stackup** dialog.

4.5 Graphic, Copper Areas

Keyboard Coordinate Input

The **Loop** option has been added to the **Jump relative** and **Jump absolute** context menu functions for creating and/or editing polygons and lines. **Loop** activates coordinate input prompts until **Abort** is selected. This allows for the keyboard input of a series of coordinate points without having to switch between mouse and keyboard.

Polygon Corner Processing

The functions from the **Polygon Corner Cut** submenu have been complemented by a procedure for recognizing and considering adjoining polygon start and end points of documentary lines as polygon corner points to allow for polygon corner manipulation functions such as corner rounding to be applied to such points.

The **Direction** and **Angle Direction** functions for changing the edit direction or for switching the angle-locked segment have been added to the context menu for creating and editing areas and lines.

The **Round Corners** button has been added to the expansion value input dialog of the **Resize Area** function. **Round Corners** ends the expansion value input and rounds the outer corners of the expanded polygon using the specified expansion value.

The **Group Arcs**, **Group 45 Degrees Angles** and **Group Eliminate Arcs** options for manipulating all group-selected polygons have been added to the **Polygon Corner Cut** function.

Arcs with Cut Segments

A redundant **Invalid polygon!** error was issued in special cases when drawing closed polygons consisting of one arc segment and a line segment at the end. This problem has been fixed.

Drawing Temporary Keep Out Areas

The **Temporary Keep Out Area** option has been added to the **Draw Rectangles** and **Draw Circles** functions of the **Drawing Utilities** submenu under **Other Functions**. The group name **temporary** is automatically assigned to keep out areas created with **Temporary Keep Out Area**. This feature is useful if board areas need to be temporarily blocked (e.g., for autorouting procedures) and then released again (e.g., for automatic copper fill procedures).

Distance Measure

The **DRELOFFSET_GED** for assigning an graphic and text distance measure offset relative to the used text size has been added to the **bae.ini** file. This is useful for displaying polygon measurement values beside and/or below the measurement line/indicator. The **DAUTOSEL_GED** entry can be set for automatically group-selecting new distance measurement elements. This allows for **Move Group** operations to be carried out on distance measurement elements immediately after their creation.

Area Mirror Display

Visible unmirrored areas as created with the **Mirror Display** function are now displayed with a highlighted dotted/pointed outline instead of a simple dotted outline to distinct them from **Visible mirrored** areas.

DXF Line Import

A feature for automatically joining subsequent lines and polygons with identical end and start point coordinates has been added to the **AutoCAD/DXF Input** function from the **Import/Export** submenu of the **File** menu.

The **Keepout Area** option for converting filled polygons into keepout areas on selectable BAE layers has been added.

The **DXFLAYASSL_LAY** entry for setting default DXF to BAE import layer assignments has been added to the **bae.ini** file.

4.6 Text, Drill

Text Size Change

New functions for increasing and reducing the text heights during the placement of texts have been assigned to the **B** (enlarge) and **S** (shrink) keys. The step value for these operations can be set through the `TEXTENLARGE_GED` parameter in the `bae.ini` file.

Project File Name Attributes

The `$pltfbname` (File Base Name) and `$pltfbsname` (File Base Short Name) system attributes for displaying the current element's project file name have been added. `$pltfbname` displays the project file name without the `.ddb` extension. `$pltfbsname` displays the project file name without the `.ddb` extension and without the directory path.

Edit Drill Parameters

The `Edit Drill` function for editing drill hole parameters such as drill hole position, drill diameter and drill class has been added to the `Text, Drill`.

4.7 Group Functions

Clipboard Data Reference Point

The `Cut` and `Copy`, when activated through `Ctrl-X` and `Ctrl-C` keys, are now using the current mouse position as reference point for the clipboard data. This allows for parts of the layout element to be copied from one (e.g., right top) corner to another (e.g., left bottom) corner of the element without having to modify the element boundaries for this placement operation. `Cut` and `Copy` still use the plan origin as clipboard reference point if called from the `Edit` menu.

Group Functions on Padstack Level

Some group functions could not be used when working on padstack level. This restriction has been removed.

Group Scaling

New functions for increasing and reducing the group scaling factor during the placement of groups have been assigned to the **B** (enlarge) and **S** (shrink) keys. The step value for these operations can be set through the `GRPENLARGE_GED` parameter in the `bae.ini` file.

Name/Attribute Positions

The `Transfer Name/Attribute Positions` function for transferring name and attribute text placement modifications from a selectable symbol to all group-selected symbols with the same macro type has been added to the `Other Functions` submenu of the `Edit` menu. This function comes in handy if specific part name and attribute text positioning needs to be applied to a group of parts such as a series of memory components.

4.8 Automatic Copper Fill

Copper Fill Parameter Assignments

A dialog for assigning area-specific copper fill parameters has been added to the **Copper Fill Area** option of the **Set Polygon Type/Net** function from the **Other Functions** submenu of the **Areas**. These parameter assignments have priority over standard copper fill parameter settings and net-specific fill parameter assignments.

It is also possible to derive the fill area net from the part pin connectivity. This ensures that the pin connections to unnamed nets are persistent if automatically generated internal net names for unnamed nets change for any reason.

Copper Fill Area Processing Sequence

The **Fill Area Priority** function for setting copper fill area processing priorities has been added to the **Other Functions** submenu of the **Areas** menu. Fill area processing priorities are considered by the **Fill all areas** function and can be used to control the copper fill sequence for nested and/or intersecting fill areas.

Copper Fill Direct Connect Pads

The new `lay_pad_directconnect` rule can be assigned to pads on padstack level to force direct copper fill connections without heat-traps for these pads, regardless of the current copper fill parameter settings.

Fill Area Via Placement

The **Dot Fill Area** function for placing via patterns to unoccupied fill area positions has been added to the **Via Functions** submenu of the **Traces** menu. Vias placed with **Dot Fill Area** are assigned to the fill area net set with **Set Fill Net** to force the system to connect these vias to the copper fill areas to be generated.

5 Autorouter

5.1 Autorouter Algorithms

Net-specific Via Padstacks

Support for net-specific via padstack definitions through `$viastk` net attribute assignments has been added to the **Autorouter**. This allows for the assignment of different via types to nets with different trace width specifications. The `tag_net*_viapadstack` tag symbols for assigning `$viastk` net attributes have been added to the **ROUTE** library.

Permanent Net Routing Disable

`$routdis` net attribute assignments can be used to permanently exclude specific nets from the autorouting process. `tag_net*_routdisable` tag symbols for assigning `$routdis` net attributes have been added to the **ROUTE** library.

Net-specific Routing Layer Assignment (BAE HighEnd)

The `$layers` net attribute can be used to assign net-specific routing layers (comma-separated signal layer numbers) for the autorouting process in **BAE HighEnd**. The `tag_net*_routlayers` tag symbols for assigning `$layers` net attributes have been added to the **ROUTE** library.

The **Special** column for assigning routing layers to nets with corresponding routing layer assignments has been added to the **Layer Assignment** dialog. This allows for routing layers to be reserved for specific nets and/or net groups.

Gridless Routing Algorithm

The **Gridless Routing 1:2** and **Gridless Routing 1:4** options for setting preferences for the placement of trace segments when routing around obstacles in gridless routing mode have been added to the grid selection menus. With these settings, trace segments are only placed gridless if no valid 1:2 or 1:4 grid position is available. In half-grid routing mode, the subgrid is in relation to the half-grid, i.e., the preferred subgrids are 1/4 or 1/8 of the routing grid, respectively. Shifting trace segments onto subgrids makes it easier to re-route them manually and reduces the possibility of minimum clearance violations potentially caused by rounding problems.

Due to numeric rounding problems, both the Rip-Up Router and the Optimizer sometimes became inefficient when coming close to gridlessly routed traces. The routing algorithms have been changed to avoid such problems, and gridless routing performance in the vicinity of gridlessly routed traces has been significantly increased.

6 CAM Processor

6.1 General

CAM Batch Output

Support for insertion data output, external command calls, **User Language** program calls, and automatic batch report output to log files has been added to the **CAM-Batch Output** function.

The element origin can now be set as CAM origin by entering an exclamation mark to the (!) origin symbol query. Previously, the lower left element boundary was used if no origin symbol was specified.

6.2 Control Plot

HPGL Default Parameter Settings

New entries for specifying default HPGL parameter settings have been added to the `bae.ini` file.

6.3 Gerber Photo Plot

Combined Layer Batch Output

The **CAM-Batch Output** function has been changed to allow for **CAM View** input layer specifications for Gerber outputs to support fully-automated multi-layer plot procedures in **CAM View**.

Dynamic Gerber Aperture Table

Due to rounding errors in the procedures for Extended Gerber output with dynamically created Gerber aperture tables, some pads suited for flashed output were filled. This problem has been solved.

7 CAM View

7.1 General

Display

A dialog has been implemented for the **Change Colors** function if called with **Color Assignment** set to **D-Code Assignment**. The D-Codes are listed pagewise with **Flash**, **Line** and **Border** buttons for direct color assignments, the basic aperture shape and the number of flash and line drawn structures being displayed with each D-Code.

7.2 Data Import

Drilling Data Import

The **Load Drill Data** function has been changed to support the import of drilling data files with different tool tables. Tool number conflicts in merged tool tables are resolved through automated tool number assignments. The **Save Tool Table** function has been added to save merged drill tool tables in Sieb & Meier format.

A dialog for selecting valid import drill classes has been added to the **Load Drill Data** function.

The **Create Layout** function has been changed to convert drill holes previously loaded with **Load Drill Data** into via padstacks with corresponding drill hole and all layer keepout areas definitions. These via padstack symbols are automatically named (**tool11**), (**tool12**), etc.

Gerber Data Import Layer Selection

The **Gerber Layer Mode** with the **Flashes & Lines on same Layer** and **Flashes & Lines extra Layers** options has been added to the **Settings** dialog from the **Settings** menu. **Flashes & Lines on same Layer** is the default setting for selecting and/or using the same destination layer for importing both line and flash structures of a specific Gerber input layer. **Flashes & Lines extra Layers** allows for the selection of different layers for line and flash structures.

Extended Gerber Aperture Type Support

Support for finger-shaped (**OBLONG**) apertures and simple aperture macros has been added to the Extended Gerber import functions.

CAM Data Import/Export Batch

The **Batch** submenu with the **Load** and **Save** functions for automatically importing and/or exporting all project-specific Gerber and drilling data files previously generated with the **CAM Processor** CAM Batch Output function has been added to the **CAM View** **File** menu. This simplifies CAM data control and CAM panelization procedures significantly.

Data Set Deletion

The **All** (default for purging all currently loaded CAM data) and **Data Set** (new feature for purging a single/selectable CAM data set) options have been added to the **Clear Memory** function.

8 Utilities

8.1 BSETUP

Pulldown Menu Configuration

The `PULLDOWN_RMB_CONTEXT` option for activating Windows/Motif pulldown menus with context menu calls through the right mouse button has been added to the `WINMENUMODE` command.

Undo Buffer

The `SCMMAXUNDO` and `LAYMAXUNDO` commands for setting the number of Undo steps in the **Schematic Editor** and the layout system have been added. The number of Undo steps must be in the range 20 to 100. The default settings are `SCMMAXUNDO (20);` and `LAYMAXUNDO (20);`.

8.2 LOGLIB

Automatic Definition Derivation

The new `autoclass` **LOGLIB** command can be used to activate a mode for automatically generating derived part definitions. The `autoclass=off;` default setting causes **LOGLIB** to compile subsequent part definitions and/or assignments as they are. With `autoclass=on;`, **LOGLIB** compiles not only the default part definition/assignment, but also generates footprint-specific sub-definitions for each layout part macro name contained in the `part` command. Automatically generated sub-definitions are named using a `symbolname_footprintname` name pattern with the symbol name automatically being used for symbol and class assignments.

The `autoclass=on;` mode is useful in connection with the `$rlext` attribute for controlling definition selections through footprint names. With `autoclass=on;` it is not necessary to redefine part assignments with identical pin assignments.

The `autoclass` mode command setting only affects subsequent definitions, thus allowing for sections with and without automatic definition derivation to be held in the same logical library definition file.

9 Bartels User Language

9.1 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 earlier BAE versions won't execute in the **User Language Interpreter** environment of the new **Bartels AutoEngineer** version (error message `User Language program version incompatible!`). This means that each **User Language** program compiled under earlier BAE Versions must be recompiled under the new **BAE** version to regain compatibility.

9.2 User Language Interpreter

This section describes the news and changes introduced to the **User Language Interpreter**. See [Bartels User Language Programmer's Guide - Chapter 3](#) for detailed information on how to operate the **User Language Interpreter**.

Implicit Program Call

A new implicit program call facility for automatically activating the **BAE_NEW** **User Language** program after creating a new DDB file element has been added to the **User Language Interpreter**.

A new implicit program call facility for automatically activating the **SCM_PLC** **User Language** program after placing a symbol has been added to the **User Language Interpreter** of the **Schematic Editor**.

9.3 Index Variable Types

This section lists the new and changed **User Language** index variable type. See [Bartels User Language Programmer's Guide - Appendix B](#) for a detailed description of all index variable types.

Changed Index Variable Types

The **NAME** attribute of the **C_FIGURE** index variable type has been changed to support symbol part name pattern queries on SCM symbol level.

9.4 System Functions

This section lists the 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_getmoduleid	Get BAE module id
	bae_numstring	Create Numeric string
	bae_peekiact	BAE interaction queue query
	bae_setmoduleid	Set BAE module id
	convstring	Convert string
	syngetintpar	Get BNF/scanner integer parameter
	synsetintpar	Set BNF/scanner integer parameter
CAP	cap_getglobnetref	Get global net name reference
	cap_getnamepat	Get SCM symbol/part name pattern
SCM	scm_getintpar	Get SCM integer parameter
	scm_pickbustap	Pick SCM bus tap with mouse
	scm_setintpar	Set SCM integer parameter
LAY	lay_getsctextdest	Get scanned layout text line destination
GED	ged_getintpar	Get GED integer parameter
	ged_setintpar	Set GED integer parameter
CV	cv_getintpar	Get CAM View integer parameter
	cv_setintpar	Set CAM View integer parameter

Changed System Functions

The [bae_swconfig](#) function has been extended to allow for BAE version build number queries.

The behaviour of the [perror](#) function for displaying error messages has been slightly changed to invert the status line output for a short moment for improved error indication. In the **User Language** programs provided with the BAE software, [perror](#) was often used for standard status messages. This has been corrected, i.e., [perror](#) calls for standard (non-error) messaging have been replaced by [bae_prtdialog](#) calls. We recommend you review the usage of [perror](#) should you deploy your own **User Language** sources/programs.

The [quickinsert](#) and [index table](#) commands for the quick insertion of large numbers of records into a database table have been added to the [sqlcmd](#) function. Unlike [insert](#), [quickinsert](#) doesn't update any database field indices. The [index table](#) command can be used after (multiple) [quickinsert](#) calls to update all field indices of a database table. This is much faster than repeatedly using the [insert](#) command which updates the table indices after each insertion. Data entered with [quickinsert](#) is not included in any query results until a [index table](#) takes place.

The [ged_groupselect](#) function has been extended by a mode for selecting and/or deselecting glued elements.

9.5 BAE User Language Programs

BAE installs more than 200 pre-compiled **User Language** programs to the `ulcprog.vdb` file of the BAE programs directory. Additionally, the **User Language** source files (more than 5 Mbytes; some 175,000 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	BAE_NEW	New Element Action
	LARGER	Increase Pick Element Size/Width
	DONE	Finish Input Interaction
	SMALLER	Reduce Pick Element Size/Width
SCM	SCM_PLC	SCM Symbol Placement Action
	SNEXTSYM	SCM Next Symbol Placement
	STXFIN	TXF Schematic Data Input
	SYMMAPDB	Create Symbol Mapping SQL Database
GED	LAYEDBAT	Layout Library Batch Editor
	LTXFIN	TXF Layout Data Input
CV	CVBATLD	Load CAM Batch Output Files

Changed User Language Programs

The **User Language** programs already delivered with the previous BAE Version have been completely revised and extended by many new features and functions. A series of significant improvements and enhancements have already been mentioned in the previous sections of these Release Notes.