TEKLA

SymEd

User's Guide

Version 2.66

Contents

2 User interface overview 15

3 Symbol files 25

4 Symbols

5 Symbol primitives

6 Bitmaps and pictures in Symbol Editor 63

Glossary 71

Index 81

Conventions used in this guide

Typographical conventions

The following typographical conventions are used in this guide:

Font	Usage
Bold	Any text that you see in the user interface appears in bold. This font is used, for example, for window and dialog box titles, box and button names, and list items.
Italic bold	New terms are in italic bold when they appear in the current context for the first time.
`Monospace`	Extracts of program code, HTML, or other material that you would normally modify in a text editor, appear in monospaced font. This font is also used for file names and folder paths, and for any text that you should type yourself.
Quotation marks	User input appears in "quotation marks". This is any text that you type into the user interface.
Arial	Names of files and directory paths appear in this font.

Noteboxes

The following types of noteboxes are used in this guide:

A **tip** might introduce a shortcut, or suggest alternative ways of doing things.

A **note** draws attention to details that you might easily over-look. It can also point you to other information in this guide that you might find useful.

You should always read very **important notes and warnings**, like this one. They will help you avoid making serious mistakes, or wasting your time.

This symbol indicates **advanced or highly technical information** that is usually of interest only to advanced or technically-oriented readers.

1   About Symbol Editor

Symbol Editor is for creating, modifying and managing symbols and symbol files in your products.

Click the links below to find out more:

Main features of Symbol Editor (p. 11)

What are primitives? (p. 12)

What are symbols? (p. 12)

What are symbol files? (p. 13)

1.1   Main features of Symbol Editor

With Symbol Editor you can do the following:

- View and modify symbol and font files.
- Create symbols with transparent primitives whose drawing attributes are specified by the product when the symbol is used at run-time.
- Fine-tune the drawing order of the symbol primitives.
- Import graphics from AutoCAD and MicroStation files.
- Use a background image as a drawing aid to draw a symbol that is available only in bitmap format.

See also   **Symbol files** (p. 25)

Modifying primitive attributes (p. 52)

Modifying transparent primitives (p. 57)

1.2   What are primitives?

Primitives are the building blocks of symbols.

A symbol is made up of *primitives* and their combinations such as lines, polygons, arcs, circles, rectangles, filled areas, and text.

Attributes
Primitives have properties such as line color, fill color or size. The properties of the primitives determine the final appearance and are defined with *attributes*. Attributes determine things such as the size, color, font and output options of a primitive. You can modify attributes through primitive-specific property dialog boxes.

Transparent attributes
The attributes can be fixed in the symbol file or they can be defined at run-time by the product. The run-time defined attributes are called *transparent attributes*. Transparency allows the product to display the symbol, for example, in different colors for different contexts on the screen.

See also

1.3   What are symbols?

Symbols are used for a variety of different purposes, for example to represent parts used in a construction assembly, to denote electrical network items such as distribution stations or utility poles; or to represent buildings, highways or parks.

You create symbols with Symbol Editor and save them in a symbol file that can contain up to 256 symbols.

You use symbols in your product. The output options of symbols may vary between different products. Basically symbols can be output on the screen, to a printer or to a file.

See also

1.4   What are symbol files?

A symbol file is a collection of individual *symbols*. A symbol file can contain up to 256 symbols. Individual symbols are identified with numbers from 0 to 255.

> Do not modify the standard symbol files delivered with your product. If you need to modify any symbols, copy the original symbol file and work on the copy and keep the original symbol file intact.

See also   **Symbol files** (p. 25)

2   User interface overview

This is an overview of the Symbol Editor user interface and its basic features.

Click the links below to find out more:

Interface components (p. 15)

Setting up the interface (p. 18)

Symbol Editor windows (p. 20)

2.1   Interface components

This is an overview of the Symbol Editor user interface components.

1. The title bar displays the Symbol Editor program name and the name and path of the currently loaded file. If no file is open at the moment, it displays the text *New file*. The menu bar contains program commands.

2. The toolbars are shortcuts to the most common file, drawing and modifying operations.

3. The **Library workarea** displays all the symbols of the current file and from here you can select a symbol to modify.

4. The **Edit workarea** is where you modify the contents of the selected symbol by inserting and modifying primitives.

5. The **Value bar** displays information related to the primitive that is selected in the **Edit workarea**. You can move or resize the primitive by typing in new coordinates in the fields of the bar and modify some primitive-specific properties.

6. On the left-hand end of the **Status bar** you can see instructions related to the drawing tool or menu command that you are using. The the right-hand end contains information related to the currently selected symbol.
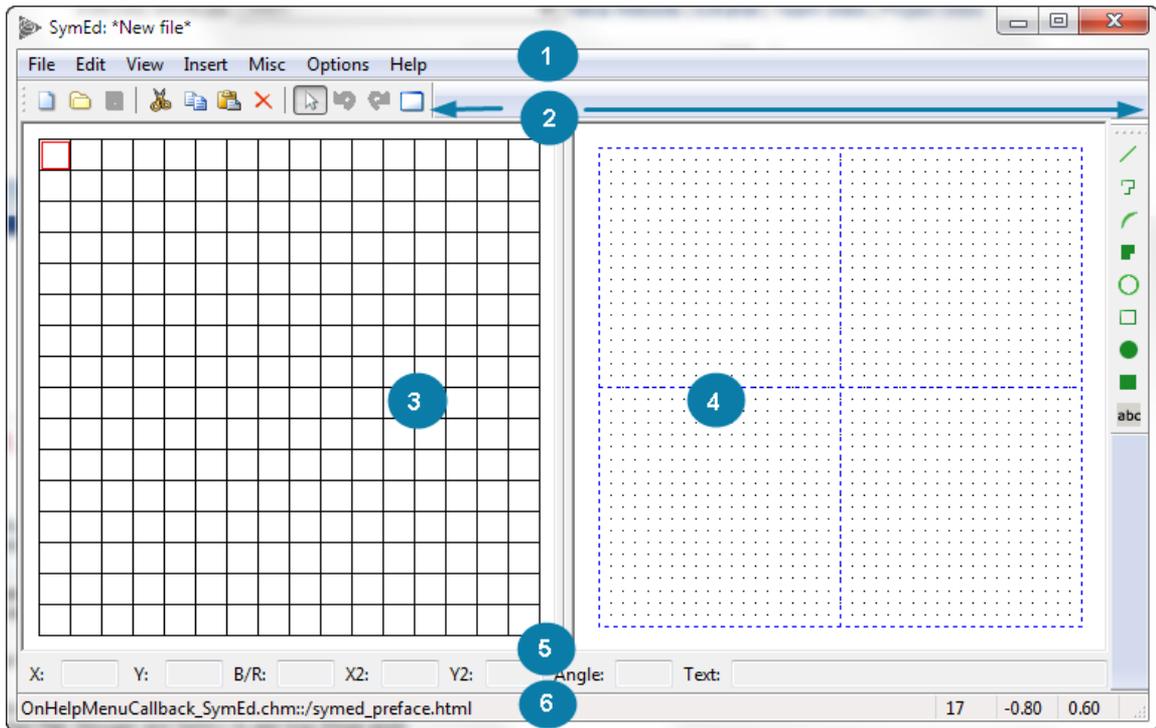
Library workarea

The **Library workarea** shows the contents of the current symbol file, so you can select a symbol to modify. There are 256 symbol slots, indexed from 0 to 255, each containing a single symbol.

Symbols are identified by the index number of the slot. Click a symbol in the library to see the index number in the **Status bar**.

See also

Edit workarea

The **Edit workarea** shows the symbol that you have selected in the **Library workarea** so you can create new primitives or modify the existing ones. The sizes and distances are measured in world units.

The size of the **Edit workarea** is 2x2 world units. The coordinate system is a regular carthesian coordinate system:

- the origin (x=0 and y=0) is in the center of the workarea
- x is the horizontal distance from the origin in world units
- y is the vertical distance from the origin in world units
- values increase up and right and decrease down and left from the origin

> The world units do not determine the final size of the symbol. Instead, they define the proportions so you can enlarge or reduce the symbol in your product with a constant scaling rate, or set the size in pixel units.
>
> For details in using symbols in your product, refer to the product documentation.

See also   **Symbol primitives** (p. 39)

Value bar

The **Value bar** appears below the workareas in the Symbol Editor main window. You can view and modify the fields that describe the location, size and some primitive-specific properties of the primitive that you have selected in the **Edit workarea**.

Field	Description
X	Sets the X-coordinate of the selected primitive's starting point.
Y	Sets the Y-coordinate of the selected primitive's starting point.
X2	Sets the X-coordinate of the selected primitive's ending point.
Y2	Sets the Y-coordinate of the selected primitive's ending point.
B/R	Applies only to lines, arcs and circles. Sets the bulge factor if a line or an arc is selected, or radius if a circle is selected.
Angle	Applies only to text primitives. Sets the rotation angle of the selected text primitive.
Text	Applies only to text primitives. Sets the text string of the selected text primitive.

Status bar

The **Status bar** appears at the bottom of the main window. As you move the mouse pointer in the main window, you see:

- tool tips for the selected command,
- the index number of the symbol that the mouse pointer is on,
- x- and y-coordinates of the mouse pointer in the **Edit workarea**.

Toolbars

You can select the most common file and modification operations from the toolbars in the Symbol Editor main window.

To view or hide the **File toolbar**, click **View > Toolbars > File**.

To view or hide the **Primitive toolbar**, click **View > Toolbars > Primitive**.

A check mark in front of the toolbar name indicates that the toolbar is visible.

2.2   Setting up the interface

This covers the program options in Symbol Editor that allow you to customize drawing options and specify the size and usage of the drawing grid. You can also set the default scaling and coordinate transformation for AutoCAD or MicroStation file imports.

Turning off automatic window updating

If the automatic redraw option is on (default setting), Symbol Editor updates all windows after each user operation. On a slow computer, this feature may cause unnecessary delays and you may want to turn it off.

To set the automatic redraw on or off, click **Options > Automatic Redraw**.

The check mark next to the option indicates that the automatic redraw is on. If you turn it off, you can update the screen manually by clicking **Misc > Redraw**.

Activating the grid

The *drawing grid* is an aid for drawing primitives in the **Edit workarea**. When the grid is active, each click snaps to the nearest grid point, so you can easily draw regular shapes.

To activate or deactivate the grid, click **Options > Grid > Active**.

A check mark in front of the option indicates that the grid is activated.

See also   **Setting grid size and pick tolerance** (p. 19)

Showing or hiding the grid (p. 19)

Snapping a primitive to the grid (p. 49)

Setting grid size and pick tolerance

To set the size of the drawing grid and the tolerance in pick operations:

1.   Click **Options > Settings** to open the **Settings** dialog box.

2.   Drag the scale pointers or type in new values for grid size and tolerance.

Set the distance between the grid points in the **Edit workarea** in world units. The distance can be 0,01-0,20 units.

Set how close to a grid point you have to click in order to snap to it. The tolerance is given in pixel units, and it can be 1-20 pixels.

3.   Click **Close**.

See also   **Activating the grid** (p. 19)

Showing or hiding the grid (p. 19)

Snapping a primitive to the grid (p. 49)

Selecting multiple primitives with one click (p. 45)

Showing or hiding the grid

Activating the grid does not automatically make it visible. On the other hand, the grid may be visible without being active.

To show or hide the grid, click **Options > Grid > Visible**.

The check mark next to the option indicates that the grid is visible.

See also   **Activating the grid** (p. 19)

Setting grid size and pick tolerance (p. 19)

2.3 Symbol Editor windows

The Symbol Editor main window displays the symbols of the currently loaded file in the **Library workarea** and if you select a symbol you can view and modify it in the **Edit workarea**. You can open one file at a time. You can create zoom windows from either one of the workareas in order to work with symbol details. This section describes working with the zoom windows.

Redrawing a window

Redrawing an active Symbol Editor window is useful if you are working on a slow computer and cannot use the automatic redraw option.

To redraw the active main window, click **Misc > Redraw** (**Ctrl + L**).

Creating a zoom window

You can create independent zoom windows for both Symbol Editor workareas. The zoom window functionality is inherited from the workarea:

*   in **Library workarea** windows you can select symbols
*   in **Edit workarea** windows you can modify symbols.

To create a new zoom window:

1. Click **View > New Zoom Window**.

2. Do **one** of the following:

    *   Drag over the area you want to view in the new window.

• Click a symbol, centering the display around the point you click.

The zoom window opens and displays the selected area or symbol.
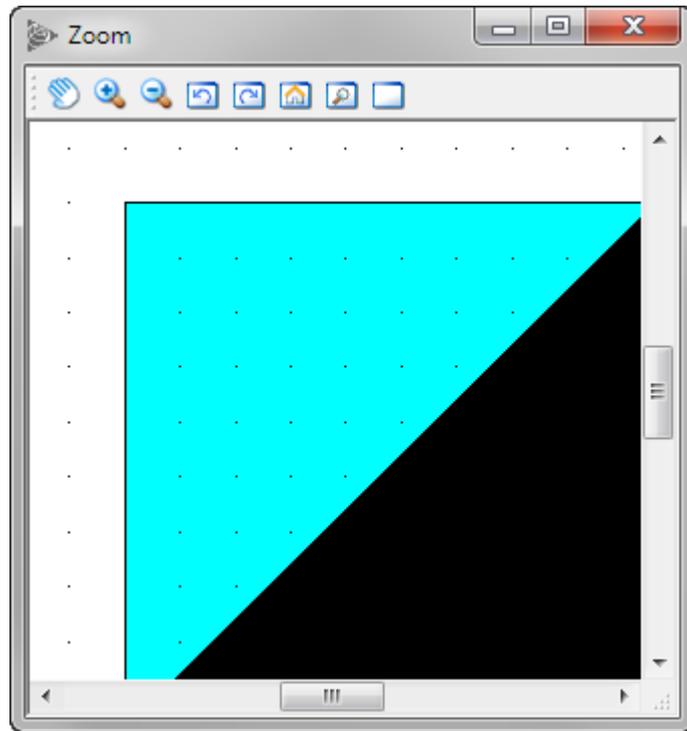
Panning

Panning is the easiest way to position the view in a window.

To pan in a zoom window:

1. Click the **Pan** button in the **Zoom** toolbar.

2. Drag in the zoom window to pan, or click to set the center point of the viewing window.

See also   **Restoring the original view** (p. 22)

Zooming in and out

To enlarge the display in the current zoom window:

1. Click the **Zoom In** button in the **Zoom** toolbar.

2. Drag over the area you want to zoom in, or click to zoom in to the next pre-set percentage. The display is centered around the point you click.

To zoom out

To reduce the display in the current zoom window:

1. Click the **Zoom Out** button in the **Zoom** toolbar.

2. Click to zoom out to the previous preset percentage.

    The display is centered around the point you click.

See also   **Restoring the original view** (p. 22)

Scrolling through the view history

Scroll through history of your actions when you need to review your work. Do the following:

- Click the **Previous View** button to go forward in the viewing history in the zoom window.

- Click the **Next View** button to go forward in the viewing history in the zoom window.

Restoring the original view

Click the **Original View** button in the zoom window to return to the original view in a zoom window.

Turning a zoom window into magnifier

You can turn a zoom window into a magnifier and magnify a part of either the **Library** or the **Edit workarea** in the zoom window.

To turn a zoom window into a magnifier:

1. Click the **Magnifier** button in the **Zoom** toolbar.

2. Move the mouse into the workarea displayed in the zoom window. The view in the zoom window changes as you move the mouse.

3. Click to end the command.

See also

You can magnify only one workarea at a time, i.e. the workarea that is displayed in this zoom window.

3 Symbol files

This is an overview of symbol files and discusses how they are created and modifed.

Click the links below to find out more:

3.1 Symbol files

You can adjust symbols for your own needs by modifying symbol files and introducing your own customized symbols without making changes in the product. Some products use symbol files directly, others use a combination of symbol and presentation files created with the Presentation Editor (PresEd). To find out how your product uses symbol files, refer to the product documentation.

3.2 Setting a font list for the symbol file

Each symbol file must define a ***font list*** that determines the fonts to use with text primitives. Symbol Editor accepts only the proprietary font files (*.FON). They are delivered with your product, for example romsim.fon and fixfont.fon. You can use up to 20 different fonts in a symbol file.

If you do not define the font list, you can use only the default font of your product. We recommend that you set the font list **before** you create symbols and use those fonts with your text primitives.

To set or modify the font list of the currently open symbol file:

1. Click **Misc > Font List**. This opens the **Font List** dialog box.
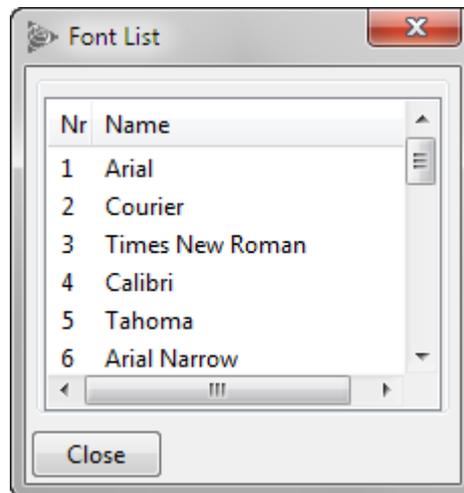
2. In the **Name** column, click on the line that you want to modify. The cursor blinks in the cell that you can type in.

   The number in the left most column is the font index.

3. Type in the name of the font, for example "romsim" or "fixfont".

   The name must be exactly the same as the base name of the corresponding font file.

4. Repeat steps 2 and 3 to define all the fonts that you want to use in symbols of the current symbol file.

5. Click **Close**.

Symbol Editor loads the fonts immediately and you can use them in your symbol.

Symbol Editor can access only the font files in the same directory as the program file SymEd.exe **or** in the directory specified with the \DXK_Fontpath environment variable.

Symbol Editor does **not** support TrueType, OpenType or PostScript font technologies.

See also   **Modifying font color, type and dimensions** (p. 55)

3.3   Creating a new symbol file

It is a rare occasion that you will create a symbol file from scratch. In case you need, do the following:

1.   Click **File > New** or click the **New** button.

This creates a new, empty symbol file. If you already have a symbol file open, Symbol Editor asks if you want to save changes to the current file.

- Click **OK** to save the changes and close the current file before creating a new file.
- Click **No** to close the current file without saving changes and to create a new file.
- Click **Cancel** to return to work with the current symbol file and cancel the creation of a new file.

2.   You can now begin to create new symbols in the empty slots of the symbol file.

See also   **Setting a font list for the symbol file** (p. 26)

Saving a symbol file (p. 28)

Creating a new symbol (p. 33)

3.4   Opening a symbol file

To open an existing symbol file:

1.   Click the **Open** button.

This opens a file selection dialog box.

2.   Select the file you want to open and click **OK**.

If you already have a file open, Symbol Editor asks if you want to save changes to the current file.

- Click **OK** to save the changes and close the file before opening the selected file.
- Click **No** to close the file without saving changes and to open the selected file.
- Click **Cancel** to return to work with the current symbol file and cancel the opening of the selected file.

> With Symbol Editor you can also view and modify font files (*.FON).

3.5 Saving a symbol file

Save a file to keep the contents intact.

To save a symbol file:

1. Click the **Save** button.

   To save a symbol file with a new name, location or format:, click **File > Save As**.

   - For a symbol file, add the extension **.SYM** after the file name.
   - For a font file, add the extension **.FON** after the file name.

2. When you save the file for the first time, the file selection dialog box opens so you can give the name and location for the file.

3. Click **OK**.

> If you specify an existing file name, Symbol Editor will over-write the existing file without a warning. Make sure all files you want to preserve intact are write-protected.

3.6 Reverting to last saved version

If you have made unwanted changes to a symbol file, you can always revert to a previous version.

To revert to the last saved version of the current file, click **File > Revert**.

This opens the previously saved version of the current symbol file. All unsaved changes are lost.

4 Symbols

This covers how to create and modify symbols.

Click the links below to find out more:

4.1 Using symbols in products

Use Symbol Editor files in products for creating user interface items. Some products (Tekla Structures, for example) use symbol files directly, others (Trimble NIS, for example) use a combination of symbol and presentation files created with *PresEd*. To find out how your product uses symbol files, refer to the product documentation.

You can adjust symbols for your own needs by modifying symbol files and introduce your own customized symbols without making changes in the product.

STOP Do not modify the standard symbol files delivered with your product. If you need to modify any symbols, copy the original symbol file and work on the copy. Keep the original symbol file intact.

4.2   Selecting items in Symbol Editor

This section covers the options in Symbol Editor that affect selecting symbols or primitives.

Extending or reducing the amount of selected items

In Symbol Editor you can use the common Windows shortcut keys to extend or reduce the amount of the selected items.

To...	Do this...
Extend the selection by selecting more items	Hold down the **Ctrl** key while selecting.
Select several subsequent items	Hold down the **Shift** key and click the first and the last item of the sequence that you want to select. This applies when selecting items in the **Primitive List**.
Deselect an item	Hold down the **Ctrl** key and click a selected item. To deselect all selected items, click outside the grid.

See also

Setting the options for selecting symbols

In Symbol Editor you can specify how to select symbols to modify in the **Library workarea**. By default you have to *double-click* a symbol, but you can also enable symbol modification after a single click by clicking **Options > Single Click Select**.

A check mark next to the menu option indicates that it is on, and you can select a symbol to modify by clicking it once in the **Library workarea**.

4.3   Creating a new symbol

To add a new symbol to the currently open symbol file:

1. Click **End Command**.

2. Double-click an empty symbol slot in the **Library workarea**.

    A red outline in the slot indicates that you can modify the symbol in the **Edit workarea**.

3. Draw the symbol by inserting primitives, importing AutoCAD or MicroStation files, or to draw a copy of an image in a bitmap file, use it as a background image in Symbol Editor.

4. To save the changes you made in the current file, click **File > Save**.

4.4   Selecting a symbol to modify

Open a symbol file to start modifying symbols.

To select a symbol to modify:

1. Click **End Command**.

2. Double-click the symbol you want to modify in the **Library workarea**.

    You can also hold down the **Shift** key and use the arrow keys to move around the **Library workarea** slot by slot.

    A red outline in the slot indicates that you can modify the symbol in the **Edit workarea**.

If you turn on the **Single Click Select** option, you will be able to select a symbol to modify with one click.

4.5   Deselecting a symbol

To deselect symbols:

1.   Click **End Command**.

2.   To deselect **all** symbols, click outside the symbol grid in the **Library workarea** or in an empty area in the **Edit workarea**.

     To deselect **one** symbol, hold down the **Ctrl** key and click the symbol slot.

Note that if you want to *change* the symbol that you are modifying in the **Edit workarea**, simply select a new symbol to modify.

4.6   Arranging symbols

Arrange symbols in the **Library workarea** when you want to reorder them according to your preferences. Note that products identify each symbol by the symbol id, which is determined by the symbol's location in the **Library workarea**. Arranging symbols also changes the symbol ids.

To arrange symbols do the following:

1.   Click **End Command**.

2.   Click the symbol that you want to move to a different slot. A black outline around the selection indicates that you can proceed.

3.   Drag the selected symbol to the desired slot. You must start dragging on top of the symbol, or you will end up making an area selection.

     If you select several symbols, other symbols are moved with the one that you drag, keeping the relations between slots intact.

4.7   Magnifying a symbol

Enlarge a symbol when you need to see it in more detail.

To enlarge the symbol view:

1.   Select the symbol you want to modify.

2.   Create a zoom window that displays the **Edit workarea**.

3.   Resize the zoom window to suit your needs.

     Take care that you can see the drawing tools of the Symbol Editor main window.

4.   Zoom in and pan the view in the zoom window to your liking.

5.   Modify the symbol in the zoom window. You can select drawing tools and menu commands from the Symbol Editor main window, and select, insert and modify primitives in the zoom window.

4.8   Cut, copy and paste

You can cut or copy symbols and primitives and paste them into another location. Symbol Editor saves the most recently cut or copied symbol in the **symbol cut buffer** and primitive in the **primitive cut buffer**. Symbol Editor maintains cut buffers until you exit the program. You can copy symbols across symbol files by opening symbol files subsequently during a Symbol Editor session.

Cutting or copying a symbol or primitive

To cut or copy a symbol or a primitive:

1.   Select the item.

2.   To cut the item, click the **Cut** button.

     To copy the item, click the **Copy** button.

     Symbol Editor adds the cut or copied item to the cut buffer, and you can paste it to another file or location.

Symbol Editor also supports the action-object model, i.e. you can take these two steps in reverse order; click **Cut** or **Copy** first, and then select the symbol or primitive.

If you want to delete an item **without** adding it to the cut buffer, click **Edit > Delete** instead of copying or cutting.

See also

Selecting items in Symbol Editor (p. 32)

Selecting a symbol to modify (p. 33)

Pasting a symbol or primitive (p. 36)

Selecting primitives (p. 45)

Pasting a symbol or primitive

You can paste a symbol or a primitive after you have cut or copied it into the cut buffer.

To paste from the cut buffer:

- To paste a symbol, click the **Paste** button, and select a target slot in the **Library workarea**. You can also select the slot first and click the **Paste** button after that.

- To paste a primitive, click the **Paste** button, and click in the **Edit workarea** to give the exact insertion point.

You can also use keyboard shortcut **Ctrl** + **V** or click **Edit > Paste** to paste items.

See also

Cut buffer (p. 37)

Cutting or copying a symbol or primitive (p. 35)

Selecting items in Symbol Editor (p. 32)

Selecting a symbol to modify (p. 33)

Selecting primitives (p. 45)

Cut buffer

All cut or copied objects (whole symbols or primitives) are stored in the Symbol Editor cut buffers. They can be retrieved in Symbol Editor with the **Paste** command. The **Library workarea** and the **Edit workarea** both have their own cut buffers that are not connected to each other. If you cut or copy a whole symbol in the **Library workarea** you can only paste it into another symbol slot. If you cut or copy individual primitives you can only paste them into the **Edit workarea**.

4.9   Deleting a symbol

Delete a symbol from a file when you no longer need it.

To delete a symbol from the currently open symbol file:

1.   Click **End Command**.

2.   Click to select a symbol in the **Library workarea**. A black outline around the selection indicates that you can proceed.

3.   Press **Ctrl + Delete**.

> If you want to move a symbol to a different file, you can cut or copy it to the cut buffer, which Symbol Editor maintains until you exit the program. Open first the source file and cut the desired symbol, then open the target file and paste the symbol.

See also   **Selecting items in Symbol Editor** (p. 32)

Cut, copy and paste (p. 35)

5 Symbol primitives

This covers inserting and modifying primitives and primitive attributes.

Click the links below to find out more:

5.1 Symbol Editor primitives

This section lists the different types of primitives and primitive groups that you can insert in symbols. Individual primitives are basic geometric shapes: lines, arcs, circles, text strings and filled areas.

Primitive groups are shapes that consist of primitives: rectangles (four lines), polylines and polygons (groups of lines) and freeforms (groups of lines and arcs).

The primitives of a symbol are listed in the **Primitive List**. Use this list as an aid while drawing and modifying primitives.

If you want to draw with the help of the grid, ensure the grid is active.

Here is a list of the primitives and any exceptions that apply to them. To insert primitives, click the related toolbar button.

Primitive	Notes
Line	Click at both ends of the line or insert by dragging.
Polyline	Move the mouse and click to define each vertex point, double-click to end drawing. The lines building the polyline or polygon appear in the **Primitive List** as separate line primitives. To modify the attributes of the polyline or polygon, select all of its lines.
Rectangle	Click to give two opposite corner points or draw by dragging. The lines building the rectangle appear in the **Primitive List** as separate primitives. To modify the rectangle, you must modify the separate lines.
Circle	Click to define the center point and click to define the radius. You can also draw by dragging.
Arc	Click at both ends of the arc, then move the mouse and click to define the bulge of the arc.
Filled rectangle	Insert the same way as a Rectangle. A filled rectangle appears in the **Primitive List** as a "Fill area".
Filled circle	Insert the same way as a circle. A filled circle appears in the **Primitive List** as a "Fill area".
Filled area	Click to define a vertex point, and continue until you have clicked all vertex points. If you want to arch the line starting from the most previously given vertex, do not click but drag to give the next vertex point. This both arches the line and inserts the new vertex. The point where you start dragging becomes the next vertex point. Click the starting point to finish drawing.
Text	Enter text into the dialog box that appears and click **OK**. Double-click to define the lower left corner of the text.

See also

Activating the grid (p. 19)

Setting a font list for the symbol file (p. 26)

Making a fill area from lines (p. 47)

Modifying the text primitive string (p. 47)

Modifying the text primitive size (p. 47)

Making lines from fill area (p. 48)

5.2   The Primitive List

The **Primitive List** displays the symbol primitives that appear in the **Edit workarea**. Symbol Editor adds new primitives at the end of the list as you draw them.

Transparent primitives are effective only when they appear at the top of the list. You can use the list as an aid when selecting primitives to modify or when arranging the transparent primitives at the top of the list after drawing them.

Symbol Editor updates the primitive list automatically when you create or delete primitives.

Displaying the Primitive List

To display the primitive list of the current symbol, click **Misc > Primitive List**.

For example, when you create a rectangle, Symbol Editor adds four line primitives to the list. The following shows an example of the **Primitive List** dialog box:
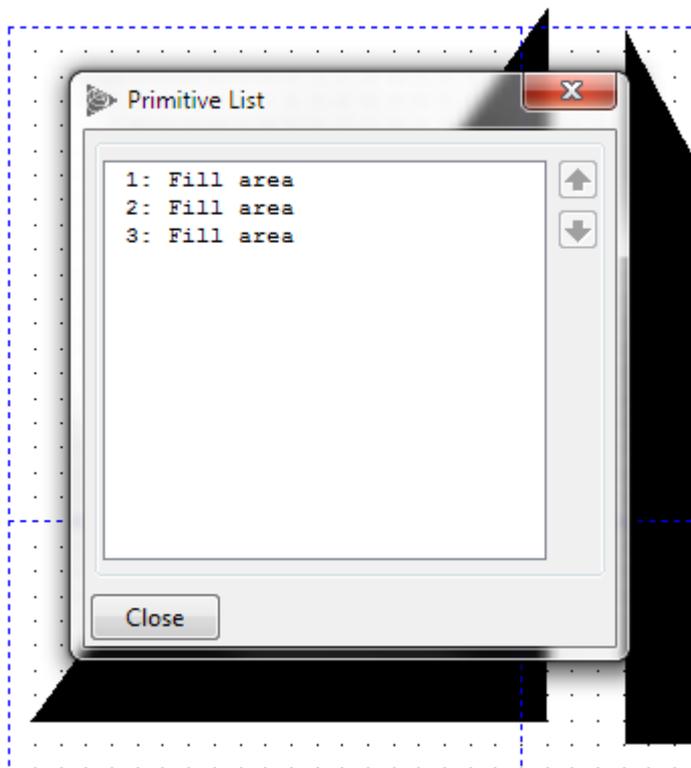
Arranging primitives in Primitive List

The order of primitives in the **Primitive List** has a significant effect on their appearance. Transparent primitives are effective only if they appear at the top of the list before primitives that have fixed attributes. Symbol Editor adds new primitives at the end of the list as you draw them so you may need to arrange their order in the list later on.

To arrange the order of primitives in the current symbol:

1. Click **Misc > Primitive List** to display the list of primitives in the current symbol.

2. Select the primitive that you want to order.

   The selection is highlighted both in the **Primitive List** and in the **Edit workarea**.

3. Order the selected primitive:

   • To move it one **step up** in the list, click **Move up**.
   • To move it one **step down** in the list, click **Move down**.

> If you select an existing primitive before you draw a new one, Symbol Editor adds the new one in the **Primitive List before** it.

See also   **Selecting primitives** (p. 45)

Opening the attributes dialog box (p. 52)

Making an attribute transparent (p. 53)

5.3   Inserting and deleting primitives

This section covers inserting and deleting primitives.

Inserting a primitive

To insert primitives in the selected symbol:

1. Select the primitive you want to insert from the Primitive toolbar.

2. Drag or click to specify the size for the primitive in the **Edit workarea**.

   Note that you can rotate line, arc and text primitives while dragging.

3. If you use both transparent and fixed primitives, arrange the **Primitive List** so that the transparent ones are at the top of the list, and fixed primitives after them.

See also   **Setting up the interface** (p. 18)

Symbol Editor primitives (p. 39)

Setting attributes for new primitives (p. 56)

The Primitive List (p. 41)

Draw several primitives running with one tool

When the continuous drawing mode is off, you can make a single drawing tool of the **Primitive** toolbar temporarily continuous. This way you can draw several primitives of the same type in succession without selecting the tool again.

1. Select a drawing tool from the **Primitive** toolbar.

2. Do **one** of the following:

    • Click **Misc > Continuous**.
    • Right-click to open the pop-up menu and select **Continuous**.

3. Draw with the drawing tool.

When you select another drawing tool, or click **End Command**, the drawing mode is discontinuous again.

See also

Draw several primitives running with all tools

You can draw several items in succession with any tool in the **Primitive** toolbar without having to reselect it when the continuous drawing mode is on.

To set continuous drawing on or off, select **Options > Continuous Drawing**.

The check mark next to the option indicates that the continuous drawing is on.

> When the continuous drawing is on, you must click **End Command** or press **Esc** to end the current command before you can select symbols or primitives to modify.

See also

5.4 Selecting primitives

This section describes how to select and deselect primitives.

Selecting multiple primitives with one click

To pick multiple primitives with one mouse click, click **Options > Pick > Multiple**.

A check mark next to the menu option indicates that the multiple pick option is on, and a single mouse click in the **Edit workarea** selects all primitives within the pick tolerance, not only the nearest one.

See also   **Selecting primitives partially inside the drag frame** (p. 45)

Setting grid size and pick tolerance (p. 19)

Selecting primitives partially inside the drag frame

To select all primitives that are either completely or partially inside the drag frame click **Options > Pick > Partially**.

A check mark next to the option indicates that the pick partially option is on, and primitives that are completely or partially inside the drag frame get selected, not just the ones that are completely inside.

See also   **Selecting multiple primitives with one click** (p. 45)

Selecting a primitive in the Edit workarea

To select a primitive:

1. Click **End Command**.

2. Click the primitive that you want to select.

    You can also drag to select several primitives inside a rectangular area.

    All selected primitives are highlighted and you can see their resize handles.

    > You can also select a primitive to modify in the **Primitive List**. This is useful, for example, when you modify a symbol that is made up of several overlapping primitives.

See also   **Extending or reducing the amount of selected items** (p. 32)

Selecting a primitive in the Primitive List (p. 46)

Selecting a primitive in the Primitive List

You can select a primitive to modify in the **Primitive List**. This is useful when you modify a symbol that is made up of several overlapping primitives. This also speeds up your work if you need to arrange the drawing order of primitives.

To select a primitive in the primitive list:

1. Click **Misc > Primitive List**. This opens the **Primitive List** dialog box.

2. Click a primitive in the list.

   You can also drag to select several subsequent primitives.

Selected primitives are also highlighted in the **Edit workarea** so you can see their resize handles.

Click **End Command** to deselect a primitive.

To deselect **all** primitives, click outside the symbol grid in the **Library workarea** or in an empty area in the **Edit workarea**.

To deselect **one** primitive, hold down the **Ctrl** key and click the primitive that you want to deselect.

See also

5.5   Modifying primitives

This section covers the different ways to modify Symbol Editor primitives.

Click the links to see more:

Making a fill area from lines

You can convert a sequence of line primitives into a filled area. This is useful, for example, when you need to group primitives that are imported from AutoCAD or MicroStation images and treat them as one object.

You can set the fill color and fill type *before* you select the command. Do the following:

1. Select the line primitives you want to convert into a filled area. Note that selected lines must form a closed shape.

2. Click **Insert > Fillarea From Lines**.

    The lines are converted into a fill area.

See also   **Setting attributes for new primitives** (p. 56)

Making lines from fill area (p. 48)

Modifying the text primitive string

To modify the text string of a text primitive:

1. Select the text primitive.

    In the value bar, click in the **Text** field and modify its contents.

2. Press **Enter** to the save changes and update the symbol.

See also   **Snapping a primitive to the grid** (p. 49)

Modifying font color, type and dimensions (p. 55)

Modifying the text primitive size

You can modify the size of a text primitive by setting the height or width of a character.

Before modifying the size you must set the scaling properties of the text primitive. It can be either *scalable* or have a *fixed pixel size*. Scalable text is scaled together with the rest of the symbol, and pixel sized text appears always in the given size regardless of the scaling of the symbol.

To modify the text primitive size do the following:

1. Select the text primitive you want to modify.

2. Click **Misc > Current Attributes**. This opens the **Selection Attributes** dialog box, where you can modify the font.

3. To set the scaling of the text, choose appropriately in the **Unit** field:

   • **World** sets the text to be scalable. This means that you will use world units to set the height and the width of text (steps 3 and 4). The characters are scaled when the symbol is scaled in a product.

   • **Pixel** sets the text to a fixed size. You will use **pixel units** to set the height and the width of text (steps 3 and 4). The characters are not scaled in a product, even though the rest of the symbol would appear scaled. Note that the view in the **Library workarea** may appear distorted: the text might not fit in the symbol slot because the view in there is scaled.

4. In the **Height** field, type in the new height to apply to the text primitive. The capital letters inherit the given height and lowercase characters are scaled accordingly.

5. In the **Width** field, type in the new width to apply to *a single character.* Each character consumes the given amount of space.

See also   **Modifying the text primitive string** (p. 47)

Snapping a primitive to the grid (p. 49)

Moving or resizing primitives with exact coordinate values (p. 50)

Making lines from fill area

You can convert a filled area to lines.

To convert a fill area to line primitives:

1. Select the fill area primitive you want to convert to line primitives.

2. Click **Insert > Lines From Fillarea**.

> You can set the line color, type and width of the new line primitives *before* you select the command.

See also   **Setting attributes for new primitives** (p. 56)

Moving a primitive by dragging

You can move primitives by dragging them, do the following:

1. Click **End Command**.

2. Select the primitive that you want to move.

3. Middle-drag to move the selection around the **Edit workarea**.

    If you drag with the *left* mouse button, you must start dragging on top of a selected object, otherwise you will make a new area selection.

See also   **Selecting a primitive in the Edit workarea** (p. 45)

Moving or resizing primitives with exact coordinate values (p. 50)

Snapping a primitive to the grid

You can move primitives onto the closest grid point in the **Edit workarea**.

1. Set grid size to suit your needs.

2. Select the primitive or primitives that you want to snap to grid.

3. Click **Edit > Snap To Grid**.

    The selected objects move onto the closest grid points.

> ⚠️ Both the starting point and the ending point are placed on the closest grid points so this may also change the size or proportions of the primitive, especially if its dimensions are smaller than the distance between grid points. Coordinates are rounded, the accuracy depends on the grid size.

See also   **Setting grid size and pick tolerance** (p. 19)

Selecting a primitive in the Edit workarea (p. 45)

Resizing the primitive by dragging

You cannot resize text primitives by dragging them.

To resize primitives:

1. Select the primitive(s) that you want to resize.

2. Hold down the **Shift** key and drag to resize.

   Note that if you use the *left* mouse button, you must place the mouse pointer on top of a selected primitive when you start dragging.

   If you use the *middle* mouse button in dragging, you can start resizing anywhere.

You see a temporary resize box. Resizing starts when you cross any of its borders and ends when you release the mouse button.

See also   **Modifying the text primitive size** (p. 47)

Moving or resizing primitives with exact coordinate values (p. 50)

Moving or resizing primitives with exact coordinate values

You can move, resize or reshape primitives by specifying exact coordinate values in world units.

You can give an exact rotation angle in degrees for text primitives.

1. Click **End Command**.

2. Select the primitive that you want to modify.

3. Type in the new coordinates in the value bar that appears right below the workareas of the Symbol Editor main window. Symbol Editor updates the primitive as you leave the field that you have changed.

See also   **Selecting a primitive in the Edit workarea** (p. 45)

Moving a primitive by dragging (p. 49)

Resizing the primitive by dragging (p. 50)

Moving or resizing primitives with exact coordinate values (p. 50)

Rotating primitives

Rotate the primitives in a symbol when you need to reposition them.

To rotate primitives:

1. Select the primitive(s) that you want to rotate.

2. Hold down the **Control** key and drag to rotate.

    Note that if you use the *left* mouse button, you must place the mouse pointer on top of a selected primitive when you start dragging. The point where you start dragging is the anchor point while rotating.

    If you use the *middle* mouse button for dragging, you can start rotating anywhere. To select the anchor point, *middle-click* it before you start to rotate by dragging.

    > You can also rotate a text primitive by typing in a new angle in the value bar that appears right below the workareas in the main window.

Modifying the shape of a primitive

You can modify the shape of a primitive by dragging the resize handles, that appear when you select the primitive. Corner handles appear in starting and ending points of the primitive, and curved lines also have a bulge point.

To modify the shape of a primitive:

1. Click a primitive to select it.

    Corner handles appear as small boxes, and bulge handles as small circles within the selected primitive.

2. Move the mouse pointer on a handle.

    Drag a *corner handle* to move a starting, ending or vertex point.

    Drag a *bulge handle* to change the curvature.

    The following figure illustrates the handles of a single straight line:

Corner handle

Bulge handle
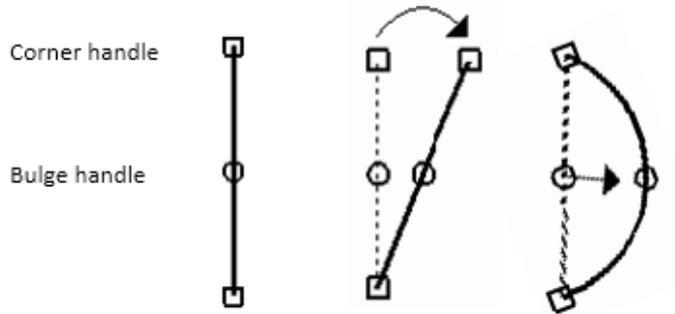
You can also modify primitive shapes by typing in new coordinates and bulge values in the value bar.

See also   **Selecting a primitive in the Edit workarea** (p. 45)

5.6 Modifying primitive attributes

Primitive attributes define the following properties of the primitive:

*   line width, color and type
*   fill color and type
*   font face, color, measurement unit, height, width, slant and thickness.

> When you modify primitive attributes, be aware that fixed attribute values override transparent attributes across primitives. You can avoid this by ordering transparent primitives at the top of the **Primitive List**.

Opening the attributes dialog box

You can modify primitive attributes by opening the **Selection Attributes** dialog box.

Do **one** of the following:

*   Double-click the primitive you want to modify.

• Select the primitive you want to modify, and select **Misc > Current Attributes**.

This opens the **Selection Attributes** dialog box.

Modify the attribute values to your liking. Symbol Editor updates the appearance of the primitive on the screen as you select new values.

See also   **Making an attribute transparent** (p. 53)

Modifying line color, type and width (p. 54)

Modifying fill color and type (p. 54)

Modifying font color, type and dimensions (p. 55)

Making an attribute transparent

To make an attribute transparent i.e. adjustable by a product:

1. Double-click the primitive you want to modify. This opens the **Selection Attributes** dialog box.

2. Set the value of the attribute that needs to be adjustable to **Default**.

When you are working in Symbol Editor, the default value is displayed with Symbol Editor's transparent attribute setting.

> *Primitives with outline* can have transparent line color, type or width. Any of the attributes may be transparent, while some of them may be fixed.
>
> *Filled primitives* can have transparent fill color and type. *Both* attributes must be transparent, if you want to use transparency with a filled primitive.
>
> *Text primitives* can have transparent font color, type and dimensions. *All* attributes must be transparent, if you want to use transparency with a text primitive.

See also   **Modifying line color, type and width** (p. 54)

Modifying fill color and type (p. 54)

Modifying font color, type and dimensions (p. 55)

Modifying transparent primitives (p. 57)

Modifying line color, type and width

You can modify the color, line type and line width of lines, arcs, polylines and circles.

1. Double-click the primitive you want to modify. This opens the **Selection Attributes** dialog box.

2. Modify the fields in the **Line** area of the dialog box.

    You can select **Default** in any field instead of using one of Symbol Editor's preset values. This makes the attribute **transparent** and thus adjustable in your product.

    The symbol is updated as you make the changes.

Field	Description
Color	Sets the line color. You can select from 8 preset colors, or use a product-specific color index.
Type	Sets the line type. You can select a solid line or a dotted line from four preset dotted lines.
Width	Sets the line width in pixels.

See also   **Defining a color with the color index** (p. 56)

Modifying fill color and type

You can modify the fill color and type of fill area primitives.

1. Double-click the primitive you want to modify. This opens the **Selection Attributes** dialog box.

2. Modify the fields in the **Fill** area of the dialog box.

    You can select **Default** in any field instead of using one of Symbol Editor's preset values. This makes *all* fill attributes **transparent**, and thus adjustable in your product.

    The symbol is updated as you make the changes.

Field	Description
Color	Sets the fill color as well as the outline color. You can select from eight (8) preset colors, or use a product-specific color index.
Type	Sets the fill type. You can select a solid fill or a fill pattern from four preset patterns.

Modifying font color, type and dimensions

You can modify the font color, type and dimensions of text primitives.

1. Double-click the text primitive you want to modify. This opens the **Selection Attributes** dialog box.

2. Modify the fields in the **Font** area of the dialog box.

    You can select **Default** in any of the first three fields instead of using one of Symbol Editor's preset values. This makes *all* text attributes **transparent**, and thus adjustable in your product.

    The symbol is updated as you make the changes.

Field	Description
Color	Sets the color of font. You can select from eight (8) pre-set colors, or use a product-specific color index.
Font	Sets the font type. You can select from the font list of the current symbol file.
Units	Defines the measurement unit to apply to this text primitive. • **World** - world units produce freely scalable texts. • **Pixel** -pixel unit texts cannot be scaled larger without losing quality.
Width	Sets the width of one character in world units or pixels, depending on the value of the **Units** field.
Height	Sets the height of text in world units or pixels, depending on the value of the **Units** field.
Slant	Sets the slanting of font, allowing you use italic type-face. Negative values (from -1 to -11) slant the text to the right.
Thickness	Sets the thickness of font in pixels, allowing you to use bold typeface.

Defining a color with the color index

Symbol Editor has eight preset colors, indexed from 0 to 7, that you can apply to line, fill or font. Some products use a wider selection of colors, with product-specific index numbers.

To set the color by using a color index, type the color index number in the field to the right of the **Color** list of the attributes dialog box.

Note that Symbol Editor may not be able to display product-specific colors correctly.

Setting attributes for new primitives

When you draw a new primitive, Symbol Editor inserts it with the current attributes that determine the line, fill or font properties.

To view and modify the current attributes:

1. *Double-click* on empty area within the **Edit workarea**.

   You can alternatively deselect all objects by clicking outside the workarea and then **Misc > Current Attributes**.

   This opens the **Current Attributes** dialog box.
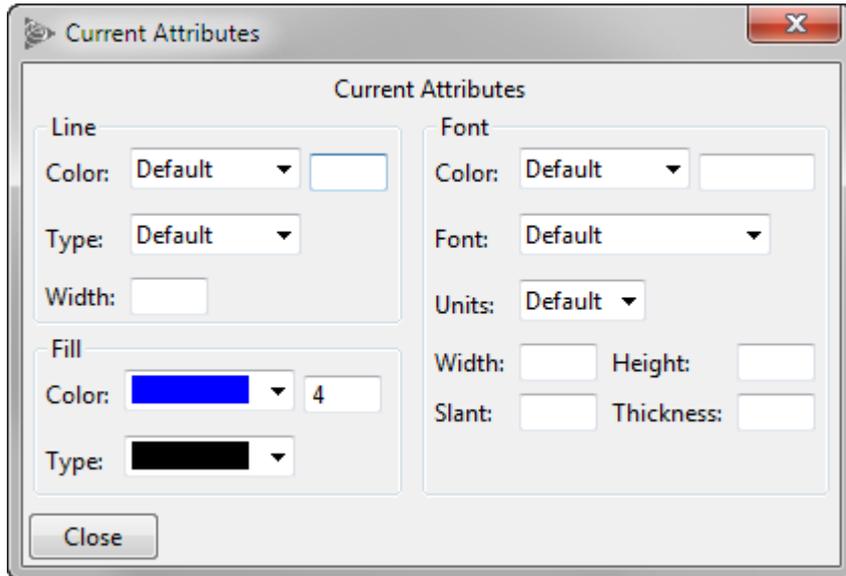
2. Modify the attribute values.

3. Click **Close** when you are done.

All new primitives are now added with the given attributes. Be aware that transparent attributes are overridden if preceding primitives in the **Primitive List** have **fixed values** for those attributes.

5.7   Modifying transparent primitives

When you insert new primitives into a symbol, Symbol Editor automatically uses *transparent* line, fill and font. This means that the product can adjust the line, fill or font properties of the symbol at run-time.

If you have modified the line, fill or font of any primitive in the symbol, or are working with an existing symbol, there may be transparency overrides so you need to make sure that transparency works the way you want in the final symbol. This section covers working with transparency.

Symbol Editor settings for transparent attributes do *not* affect the final appearance of symbols. products adjust and determine the final appearance, when they use symbols that have transparent attributes.
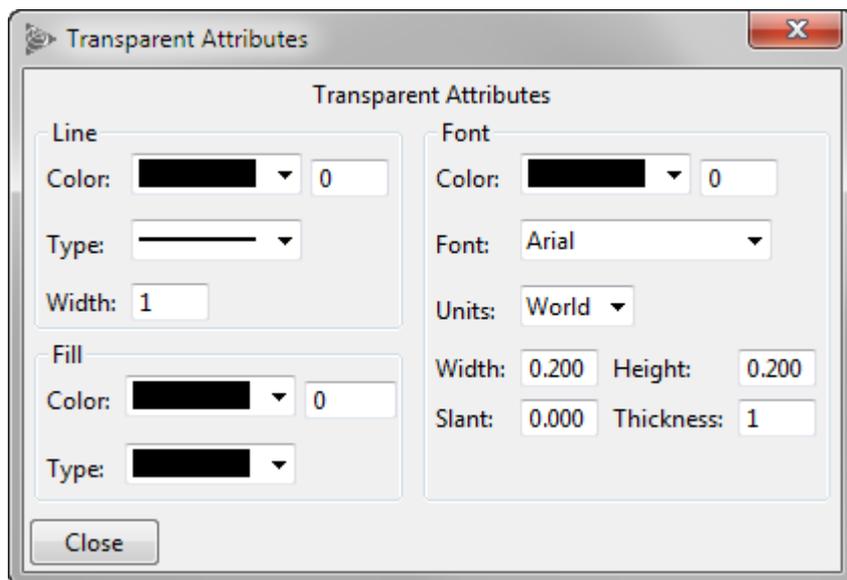
Setting appearance of transparent attributes

Symbols that have transparent attributes can be adjusted by your product at run-time. You can test how the symbol will appear at run-time, when you change the way transparent attributes, (i.e.) adjustable line, fill and font, appear in Symbol Editor.

To change the way transparent line, fill and font show in Symbol Editor:

1. Click **Misc > Transparent Attributes**.

   This opens the **Transparent Attributes** dialog box.

2. Modify the attribute values. Symbol Editor updates the appearance of primitives on the screen as you select new values.

   The transparent attribute settings are Symbol Editor's internal settings so they do **not** determine the appearance of the final symbol but assist you when making primitives transparent and testing how it works. Your product sets the final appearance of transparent line, fill and font at run-time.

See also   **Troubleshooting transparency problems** (p. 60)

   Making an attribute transparent (p. 53)

   Modifying line color, type and width (p. 54)

   Modifying fill color and type (p. 54)

Making a primitive transparent

You can make a primitive transparent and thus adjustable in your product when the symbol is used in it. Do the following:

1. Decide which attributes need to be adjustable:

   - *Primitives with outline* can have transparent line color, type or width. Any of the attributes may be transparent, while some of them may be fixed.
   - *Filled primitives* can have transparent fill color and type. *Both* attributes must be transparent, if you want to use transparency with a filled primitive.
   - *Text primitives* can have transparent font color, type and dimensions. *All* attributes must be transparent, if you want to use transparency with a text primitive.

2. Modify the primitive's attributes to make the selected attributes transparent.

See also   **Making an attribute transparent** (p. 53)

Removing transparency from a primitive

Modify the transparency of a primitive when you want to change how it appears at run-time.

To remove transparency of a primitive and fix its properties in Symbol Editor:

1. Highlight the transparent attributes.

2. Select the primitive whose line, fill or font you want to fix so that the product cannot change it at run-time.

3. Fix the line, fill or font by selecting a pre-defined line, fill or font.

4. Save the changes to the symbol file.

> If transparency of the other primitives does not work the desired way after this operation, you will have to troubleshoot the problem.

See also   **Modifying line color, type and width** (p. 54)

Highlighting transparent primitives

You can use any color or style to highlight the transparent primitives of a symbol in Symbol Editor. Highlighting does not affect the final symbol in any way but makes it easy for you to check how transparency works.

1. Select the symbol you want to modify.

2. Modify Symbol Editor's internal settings for transparent line, fill and font so that they differ from the current appearance of your symbol as much as possible.

   For example, if you want to check which primitives have transparent fill settings in a symbol that currently appears in black, choose a bright fill color that makes the transparent fill area primitives stand out. Now you can concentrate on working with the fill areas that do not act they way you want them to.

See also   **Setting appearance of transparent attributes** (p. 58)

Troubleshooting transparency problems

If symbol transparency does not work in the way that you expected, the problem is most likely that a fixed line, fill or font attribute is overriding the transparency.

To remove the overrides and restore the effectiveness of the transparent primitives:

1. Highlight the transparent primitives of the symbol.

2. Make sure that all the primitives with transparent attributes are **at the top** of the **Primitive List**.

Be aware that transparency overrides can also occur **across** partially transparent primitives. A partially transparent primitive has both transparent and fixed attributes. For example, a line primitive can have a transparent line color while its line width is fixed. Any primitives that have transparent line width will be drawn with this fixed width, if they appear below this line primitive in the **Primitive List**. To avoid this, list primitives with solely transparent attributes topmost, and primitives that have both transparent and fixed attributes after them. Primitives that only have fixed attributes should appear last in the **Primitive List**.

See also

Symbol primitives

6

Bitmaps and pictures in Symbol Editor

This covers how to use graphics with other file types for symbols. You can import AutoCAD or MicroStation files, use bitmap pictures as a drawing aid in the background of the **Edit workarea**, and use a digitizer tablet to import pictures available only as paper copies.

Click the links below to find out more:

Importing AutoCAD or MicroStation files (p. 64)

Using a bitmap file as a background image (p. 65)

Using a digitizer tablet (p. 66)

6.1   Importing AutoCAD or MicroStation files

You can use AutoCAD (DXF or DWG) files and MicroStation (DGN) files in Symbol Editor. Symbol Editor imports the files and converts their contents into *primitives* that you can modify.

To import a file:

1. Click **Options > Import transform** to set the scaling and the initial positioning of the file to import.

2. Enter the values for scaling the imported image and shifting the x and y coordinates.

   - **Scale**: Type in the scaling rate. A value of 1 means that the image is imported in its original size. A value of 2 means that the image is imported twice as large as the original.
   - **Translate x**: Shifts the imported image in a horizontal direction. Negative values shift the image to the left, and positive values to the right. Values are in world units.
   - **Translate y**: Shifts the imported image in a vertical direction. Negative values shift the image down, and positive values up. Values are in world units.

   If the value of both translation fields is 0, the origin of the imported image is placed on the origin of the **Edit workarea**.

3. Click **OK**.

4. In the **Library workarea**, select the symbol into which you want to import the image.

5. Click **File > Import**. This opens the standard file selection dialog box.

6. Select the file to import and click **OK**.

The image is converted to Symbol Editor primitives that are inserted into the selected symbol. Scaling and positioning are set according to the settings set in step 2.

To check how the objects from the AutoCAD or MicroStation file are mapped to Symbol Editor primitives see the next section.

> You can group the primitives of the image and handle them as one object.

See also

6.2   Mappings and limitations of file imports

This describes how AutoCad and MicroStation files are converted into Symbol Editor drawing primitives.

Object mappings   The following object types are imported from the original file to the symbol: ARC, LINE, CIRCLE, CELL, CURVE, SHAPE, SOLID, TEXT, TRACE, INSERT, POLYLINE, LWPOLYLINE, LINESTRING and HATCH.

Fill type mappings   The fill type of a filled object is converted as follows:

Fill style name in DWG/DXF/DGN file	Fill type in Symbol Editor
NONE	transparent
SOLID	solid fill
SINGLEHATCH	diagonal lineation
DOUBLEHATCH	diagonal lineation
all other fill types	transparent

Text limitations   The fonts used in text objects are mapped to Symbol Editor fonts. This may slightly distort the font alignment within the text box as the original font geometry is not available in Symbol Editor.

Other limitations   User-defined line styles and world unit line widths from AutoCAD files are not imported.

6.3   Using a bitmap file as a background image

You can use a background image in Symbol Editor as a drawing aid to draw a symbol that is available only in bitmap format. You cannot import bitmap files.

Bitmap files Symbol Editor can display as background image

Click **File > Background Image** to choose a background image.

A file selection dialog box opens for selecting a bitmap file (GIF, JPEG or TIF) that you can use as a drawing aid to draw a symbol that is available only in bitmap format.

The bitmap image will appear as a background for all symbols in the current symbol file in the modification phase, but it does not show in the final symbols.

Setting a bitmap file as the background image

To use a background image:

1. Click **File > Background Image**. This opens the standard file selection dialog box.

2. Select the file you want to use as a background image and click **OK**.

    The bitmap is displayed in the background of **Edit workarea**, and it appears there for all the symbols of the current symbol file.

    You can control the visibility of the background image with a Symbol Editor program option.

    > The background bitmap does **not** appear in final symbols. It is only a help feature for drawing symbols from bitmap files.

See also   **Show or hide background image** (p. 66)

Show or hide background image

If you have selected a background for a symbol file, it appears in the **Edit workarea** for all the symbols of the file.

To show or hide the background image, select **Options > Background Image**.

A check mark next to this menu option indicates that the background image is visible

Note that the background does **not** appear in zoom windows, regardless of this visibility setting.

6.4   Using a digitizer tablet

You can transfer a paper copy into digital format with a digitizer tablet. You can for example digitize your company logo from a paper copy. Lay the paper copy on the digitizer tablet and the dimensions are transferred to Symbol Editor as you trace them.

Gathering digitizer tablet connection data

Before connecting a digitizer tablet to Symbol Editor, make sure it is connected to your computer and look for the following connection data:

- number of the communications port, typically COM1 or COM2
- connection speed
- number of stop bits
- number of data bits
- parity

See also   **Connecting a digitizer tablet to Symbol Editor** (p. 67).

Connecting a digitizer tablet to Symbol Editor

You connect a digitizer tablet to Symbol Editor when you need to re-draw a symbol from a printed source.

To connect a digitizer tablet to Symbol Editor, do the following:

1. Gather all of the connection data you need.

2. Connect the digitizer tablet to your computer.

3. In Symbol Editor, select **File > Digitizer**. The **Digitizer String** dialog box opens prompting you for the connection data.

4. In the **Format String** field, enter a string that contains the following information in the following order:

   - COM port number (COM1 or COM2)
   - baud rate speed (by default 9600)
   - number of data bits (by default 8)
   - number of stop bits (by default 1)
   - parity (by default 0)
   - an ASCII number (typically 10 or 13)
   - the string "NNKXXXXXYYYYY"

   Separate the COM port, speed, data bit, stop bit, and parity information with **colons**. Separate the ASCII number and string information with **commas**.

   The format string may look as follows:

   `COM1:9600:8:1:0,13,NNKXXXXXYYYYY`

5. In the **Button 1 Char(s)** field, enter the number of the button on the digitizer mouse device that you want to map to the left mouse button.

6. In the **Button 2 Char(s)** field, enter the number of the button on the digitizer mouse device that you want to map to the middle mouse button.

Bitmaps and pictures in Symbol Editor

7. In the **No Button Char(s)** field, enter the character that Symbol Editor inter-prets as input from the digitizer when the digitizer is in *streaming* (continu-ous) mode, and no buttons on the digitizer mouse device are pressed. If you leave the field empty, no input is received from the digitizer.

8. Click **OK**.

Symbol Editor asks you to define the drawing area of the digitizer by click-ing on the lower left and upper right corner of the digitizer tablet. After that, the digitizer is ready for use.

See also　　

Connecting digitizer tablet to Symbol Editor using environment variables

This section is for advanced users and lists the environment variables that you can use to specify the digitizer connection to Symbol Editor. You can modify these variables by modifying your system's environment variables.

1. Define the connection with the environment variables listed in the table below.

Variable	Value	Explanation
SYMEDDIGIT	String	Sets the format string. (See the previous sec-tion for more information).
SYMEDDIGIT0	Character	Sets the character that Symbol Editor inter-prets as input from the digitizer when the digitizer is in streaming (continuous) mode, and no buttons on the digitizer mouse device are pressed.
SYMEDDIGIT1	Digitizer mouse but-ton	Maps the left mouse button to a specified digitizer mouse button. For example, to map the left mouse button to the digitizer mouse button number 0: `SYMEDDIGIT1=0`
SYMEDDIGIT2	Digitizer mouse but-ton	Maps the middle mouse button to a speci-fied digitizer mouse button. For example, to map the middle mouse button to the digi-tizer mouse button 3: `SYMEDDIGIT2=3`

2. Start digitizing by selecting **File > Digitizer**.

   Symbol Editor prompts you to define the digitizer drawing area by clicking on the lower left and upper right corner of the digitizer tablet. After that, the digitizer is ready for use.

See also

Digitize an image from a paper copy with a digitizer tablet

You can digitize an image with a digitizer tablet when you have a large-scale paper copy of the intended symbol. Do the following:

1. Make sure the digitizer tablet is connected to your computer.

2. In the **Library workarea**, select the symbol where to draw the image from the digitizer tablet.

   You can change the symbol at any phase of digitizing.

3. Place the paper copy of the symbol on the digitizer tablet.

4. Digitize the symbol using the digitizer's mouse device. A digitizer mouse is used in the same way as a computer mouse.

For more information about using a digitizer tablet, see your digitizer tablet documentation.

See also

Bitmaps and pictures in Symbol Editor

Glossary

A

Arc

A curved line defined by two end points and a bulge factor. One of the primitives used to compose symbols in Symbol Editor.

Attribute

Attributes determine primitive's line, fill and font properties. Attributes can be fixed in Symbol Editor or they can be left transparent, in which case the product that uses the symbol at run-time defines the final appearance of line, fill or font.

B

Background image

A bitmap image (GIF, JPEG, or TIF) displayed in the **Edit workarea**. You use a background image to facilitate creating a symbol with a similar shape. For example, if you have your company logo in a GIF file, you display it as a background image and trace its edges to reproduce the shape using Symbol Editor primitives.

Baud rate

A measure of signaling speed in data communications that specifies the maximum number of signal elements transmitted each second.

Bulge factor

A fractional number that defines an arc between two points as x/y, where y is the distance between the points and x is the distance between the center of the arc and the center of the imagined straight line between the points. A positive bulge factor produces an anti-clockwise arc. A negative bulge factor produces a clockwise arc. The shape of the arc is always part of a complete circle.

Bulge handle

A circular symbol displayed in the middle of a line or an arc primitive when you select the primitive. You can drag the handle to modify the bulge of an arc primitive. If you drag the bulge handle of a line primitive, it changes to an arc.

Bulge point

The middle point of an arc primitive. When you select an arc, a bulge handle is displayed in the bulge point.

C

Corner handle

A square symbol displayed at certain points of a primitive when you select it. You can drag a corner handle to change the shape of the corresponding primitive. For example, you can drag a corner handle displayed at the end of a line primitive to move the corresponding end point, thereby modifying the length and direction of the line.

Current area

The rectangular area that is currently visible in the **Edit workarea**.

Current attributes

Attributes of the currently selected primitive i.e. properties of line, fill or font. Attribute values can be fixed or transparent:

- A fixed attribute value is specified in Symbol Editor. It applies to the selected primitive and all the primitives below the selected one in the **Primitive List** down to a primitive that defines another fixed value.

- A *transparent attribute* is specified by product when it uses the symbol. In Symbol Editor transparent attributes appear in Symbol Editor default values that do not have an effect on the final symbol. Fixed values override transparent values.

Cut buffer

All cut or copied objects (whole symbols or primitives) are stored in the Symbol Editor cut buffers. They can be retrieved in Symbol Editor with the **Paste** command. The **Library workarea** and the **Edit workarea** both have their own cut buffers that are not connected to each other. If you cut or copy a whole symbol in the **Library workarea** you can only paste it into another symbol slot. If you cut or copy individual primitives you can only paste them into the **Edit workarea**.

D

DGN

MicroStation file format.

Default attribute value

If you select the **Default** value for an attribute instead of using one of Symbol Editor's preset values or a product-specific value, you make the attribute transparent, and the product determines the value for this attribute at run-time.

Digitizing

The process of converting manually prepared source materials into the digital codes stored and processed by computers.

Digitizer tablet

A hardware device that allows you to transform an image printed on a piece paper into a vector-based computer image.

Drawing primitive

An undivided unit of a graphic presentation. Symbol Editor uses five drawing primitives: line, arc, circle, filled area, and text.

DXF

Native CAD file format. Basic shapes of CAD applications can be imported into Symbol Editor in DXF format.

DWG

AutoCAD file format.

E

Edit workarea

The right hand part of the Symbol Editor main window that shows the parts of the selected symbol so that you can modify it.

Ending point

The definition of the ending point depends on the primitive type:

- **Line**: where you click to define the end point.
- **Text**: where you click to define the end point.
- **Rectangle**: upper right corner point of the rectangle.
- **Area**: upper right corner point of a rectangle that the area fits into.

F

FON

Font file format.

G

No entries beginning with G.

H

No entries beginning with H.

I

No entries beginning with I.

J

No entries beginning with J.

K

No entries beginning with K.

L

Library workarea

The left hand part of the Symbol Editor main window that shows the contents of the symbol file, made up of up to 256 separate symbols.

M

No entries beginning with M.

N

No entries beginning with N.

O

No entries beginning with O.

P

Pixel

A measurement unit that is fixed by the size and resolution of the screen of your computer.

Polygon

A multi-sided figure representing an area.

Presentation

A collection of drawing instructions that use Symbol Editor symbols, so you can define complex graphic presentations for products. You create and modify presentations with PresEd.

Primitive

A basic vector shape such as line, circle, or text string. Symbol Editor symbols are made up of primitives. Polylines, polygons, rectangles and freeforms are not primitives but collections of line and arc primitives laid out in a certain pattern. Filled areas of those shapes are considered individual primitives.

Primitive List

A list showing the primitives of the current symbol as a stack. The order of primitives in that list affect their appearance because fixed attributes override transparent (i.e. product-defined) attributes. To avoid this, primitives with fixed attribute values should be in the **Primitive List** below primitives that have transparent attribute values.

Q

No entries beginning with Q.

R

No entries beginning with R.

S

Starting point

The definition of the starting point depends on the primitive type:

- **Line**: where you click to define the starting point.
- **Text**: where you click to define the starting point.
- **Rectangle**: lower left corner point of the rectangle.
- **Circle**: center point.
- **Area**: lower left corner point of a rectangle that the area fits into.

SYM

Symbol file format.

Symbol

In Symbol Editor, a basic vector image that consists of primitives. Symbols are in symbol files, which are used by products.

Symbol file

A file that contains up to 256 individual symbols. The file has the extension SYM.

T

Transparent attributes

Attributes whose values are specified by the product when using the symbol at run-time. In Symbol Editor transparent attributes appear in Symbol Editor as a default value, which does not affect the final symbol. Transparent attributes are effective only when the primitives with transparent attributes appear in the **Primitive List** before primitives with fixed attributes.

Transparent primitive

A primitive is transparent with respect to a given attribute if it does not fix a value for that attribute in Symbol Editor. The actual value of that attribute is defined at run-time by the product that uses the symbol. The transparent primitives are effective only when they appear in the **Primitive List** before primitives with fixed attributes.

U

No entries beginning with U.

V

No entries beginning with V.

W

World coordinate system

The coordinate system in the **Edit workarea**. The coordinate values grow from left to right (x) and from bottom up (y) and the values are between -1 and 1.

World units

The coordinate units in the **Edit workarea**. You can define the dimensions (width and height) of symbols in world units instead of fixing the size in pixel units, millimeters or inches. The product that uses the symbols defines the actual coordinate unit (for example millimeters or inches) to be used and scales the symbols for display.

X

X coordinate

The X coordinate determines the position of the primitive in the horizontal direction. Positive values place the primitive to the right of the origin, negative values to the left.

Y

Y coordinate

The Y coordinate determines the position of the primitive in the vertical direction. Positive values place the primitive above the origin, negative values below it.

Z

No entries beginning with Z.

Index