

# BasicCAD Changes and Additions

## V18-24.1

---

### BasicCAD Changes in Version 18

#### New Query/Change parameters for dimension entities:

##### D\_Prefix

Declaration: D\_PREFIX &H0A0D

Arguments: A string value indicating the dimension prefix text.

Accessibility: Query/Change

Purpose: This parameter allows the programmer to determine a dimension prefix. If parameter is presented as empty string then no prefix appears for that dimension. NOTE: This allows you to directly specify the prefix text, rather than selecting it from a list.

Query Usage:

Query D\_Prefix, dstr\$  
where dstr is a string variable.

Change Usage:

Change D\_Prefix, dstr\$  
where dstr can be a string variable or expression.

Compatibility: Version 18 and higher

##### D\_Suffix

Declaration: D\_SUFFIX &H0A0E

Arguments: A string value indicating the dimension suffix text.

Accessibility: Query/Change

Purpose: This parameter allows the programmer to determine a dimension suffix. If parameter is presented as empty string then no suffix appears for that dimension.. NOTE: This allows you to directly specify the prefix text, rather than selecting it from a list.

Query Usage:

Query D\_Suffix, dstr\$  
where dstr is a string variable.

Change Usage:

Change D\_Suffix, dstr\$  
where dstr can be a string variable or expression.

Compatibility: Version 18 and higher

### **D\_ToleranceType**

Declaration: Alias D\_TOLERANCETYPE &H0A0F

Arguments: A numeric value (integer [0-2]) indicating the dimension tolerance type. Values: 0 = no tolerance, 1 = symmetrical tolerance, 2 = normal tolerance.

Accessibility: Query/Change

Purpose: This parameter allows the programmer to determine a type of dimension tolerance.

Query Usage:

Query D\_ToleranceType, dtype  
where dtype is a numeric variable.

Change Usage:

Change D\_ToleranceType, dtype  
where dtype can be a numeric variable or expression.

Compatibility: Version 18 and higher

### **D\_ToleranceHigh**

Declaration: Alias D\_TOLERANCEHIGH &H0A10

Arguments: One floating-point value representing the dimension upper tolerance.

Accessibility: Query/Change

Purpose: This parameter allows the programmer to determine an upper tolerance.

Query Usage:

Query D\_ToleranceHigh, dtoler  
where dtoler is a numeric variable.

Change Usage:

Change D\_ToleranceHigh, dtoler  
where dtoler can be a numeric variable or expression.

Compatibility: Version 18 and higher

### D\_ToleranceLow

Declaration: Alias D\_TOLERANCELOW &H0A11

Arguments: One floating-point value representing the dimension lower tolerance.

Accessibility: Query/Change

Purpose: This parameter allows the programmer to determine a lower tolerance.

Query Usage:

Query D\_ToleranceLow, dtoler  
where dtoler is a numeric variable.

Change Usage:

Change D\_ToleranceLow, dtoler  
where dtoler can be a numeric variable or expression.

Compatibility: Version 18 and higher

### D\_ToleranceSize

Declaration: Alias D\_TOLERANCESIZE &H0A12

Arguments: One floating-point value representing the dimension tolerance size scale.

Accessibility: Query/Change

Purpose: This parameter allows the programmer to determine scale factor between normal tolerance size and dimension text size.

Query Usage:

Query D\_ToleranceSize, dtoler  
where dtoler is a numeric variable.

Change Usage:

Change D\_ToleranceSize, dtoler  
where dtoler can be a numeric variable or expression.

Compatibility: Version 18 and higher

### **D\_TextLocation - deleted**

***This Query/Change parameter (formerly aliased to &H0A14) has been deleted from DesignCAD Version 18 and higher.*** New macro code should use D\_TextPosition\_Ver (below) and D\_TextPosition\_Hor (below) to choose the text position.

### D\_TextPosition\_Ver

Declaration: Alias . D\_TEXTPOSITION\_VER &H0A14

Arguments: A numeric value (integer [0-2]) indicating the vertical position of the text relative to the arrows. This parameter is only used when Fixed Text Position is enabled. Values: 0=centered between arrows, 1=above the arrows, 2=below the arrows

Accessibility: Query/Change

Purpose: This parameter allows the programmer to determine or change a dimension's vertical text position.

Query Usage:

Query D\_TextPosition\_Ver, textpos  
where textpos is a numeric variable.

Change Usage:

Change D\_TextPosition\_Ver, textpos  
where textpos can be a numeric variable or expression.

Compatibility: Version 18 and higher

### **D\_TextPosition\_Hor**

Declaration: Alias D\_TEXTPOSITION\_HOR &H0A15

Arguments: A numeric value (integer [0-4]) indicating position of text (if FixedTextPosition is enabled). Values: 0=centered, 1=left, 2=right, 3=over left extension line, 4=over right extension line

Accessibility: Query/Change

Purpose: This parameter allows the programmer to determine or change a dimension's horizontal text position.

Query Usage: Query D\_TextPosition\_Hor, textpos where textpos is a numeric variable.

Change Usage: Change D\_TextPosition\_Hor, textpos where textpos can be a numeric variable or expression.

Compatibility: Version 18 and higher

### **D\_ArrowType2**

Declaration: Alias D\_ArrowType2 &H0A16

Arguments: A numeric value (integer [0 - 13]) indicating the dimension second arrowhead type.

Accessibility: Query/Change

Purpose: This parameter allows the programmer to determine or change a dimension entity's second arrowhead type.

Query Usage: Query D\_ArrowType2, artype2 where artype2 is a numeric variable.

Change Usage: Change D\_ArrowType2, artype2 where artype2 can be a numeric variable or expression.

Compatibility: Version 18 and higher

### **D\_EnableSecondArrow**

Declaration: Alias D\_EnableSecondArrow &H0A17

Arguments: A number (integer [0, 1]) indicating whether second arrow type is used or no. Values: 0 = both arrow types are defined by first arrow type, 1 = different arrows

Accessibility: Query/Change

Purpose: This parameter allows you to determine if a dimension is using a second arrow type or second arrows is the same as first one.

Query Usage: Query D\_EnableSecondArrow, dsecondarrow where dsecondarrow is a numeric variable.

Change Usage: Change D\_EnableSecondArrow, dsecondarrow where artype2 can be a numeric variable or expression.

Compatibility: Version 18 and higher

### **D\_EnableFixedTextPos**

Declaration: Alias D\_ENABLEFIXEDTEXTPOS &H0A18

Arguments: A number (integer [0, 1]) indicating whether text position is defined by D\_TextPosition\_Ver and D\_TextPosition\_Hor or by last point. Values: 0 = text position is defined by last point, 1 = text position is fixed

Accessibility: Query/Change

Purpose: This parameter allows you to determine if a text position of dimension is fixed or no.

Query Usage: Query D\_EnableFixedTextPos, dtextpos where dtextpos is a numeric variable.

Change Usage: Change D\_EnableFixedTextPos, dtextpos where dtextpos is a numeric variable or expression.

Compatibility: Version 18 and higher

### **D\_TextOrientation - deleted**

***This Query/Change parameter (formerly aliased to &H0A15) has been deleted from DesignCAD Version 18 and higher. New macro code should use D\_SetTextFlipped (below).***

### **D\_SetTextFlipped**

Declaration: Alias D\_SETTEXTFLIPPED &H0A19

Arguments: A number (integer [0, 1]) indicating whether is text flipped or no. Values: 0 = text is not flipped,, 1 = text is flipped

Accessibility: Query/Change

Purpose: This parameter allows you to determine if a text position of dimension is flipped or no.

Query Usage: Query D\_SetTextFlipped, dtextflipped where dtextflipped is a numeric variable.

Change Usage: Change D\_SetTextFlipped, dtextflipped where dtextpos is a numeric variable or expression.

Compatibility: Version 18 and higher

### **D\_KeepTextInside**

Declaration: Alias D\_KEEPTXTINSIDE &H0A1A

Arguments: A number (integer [0, 1]) indicating whether is text always kept inside extension lines for center horizontal alignment with fixed text position or no. Values: 0 = text can be outside, 1 = text is kept inside

Accessibility: Query/Change

Purpose: This parameter allows you to determine if a text is always kept inside extension lines for center horizontal alignment with fixed text position or no.

Query Usage: Query D\_KeepTextInside, dkeep where dkeep is a numeric variable.

Change Usage: Change D\_KeepTextInside, dkeep where dkeep is a numeric variable or expression.

Compatibility: Version 18 and higher

### **D\_DrawArrowsOutside**

Declaration: Alias D\_DRAWARROWSOUTSIDE &H0A1B

Arguments: A number (integer [0, 1]) indicating whether are arrows always outside extension lines or no. Values: 0 = arrows can be inside, 1 = arrows are always outside

Accessibility: Query/Change

Purpose: This parameter allows you to determine if arrows are always drawn outside extension lines or no.

Query Usage: Query D\_DrawArrowsOutside, doutsidearr where doutsidearr is a numeric variable.

Change Usage: Change D\_DrawArrowsOutside, doutsidearr where doutsidearr is a numeric variable or expression.

Compatibility: Version 18 and higher

### **D\_Units**

Declaration: Alias D\_UNITS &H0A1C

Arguments: A number (integer [-1, 7]) indicating dimension units. Values: -1 = no units(multiplier is enabled), 0 = Unitless, 1 = Inches, 2 = Feet, 3 = Miles, 4 = Millimeters, 5 = Centimeters, 6 = Meters, 7 = Kilometers

Accessibility: Query/Change

Purpose: This parameter allows you to determine dimension units.

Query Usage: Query D\_Units, dunits where dunits is a numeric variable.

Change Usage: Change D\_Units, dunits where dunits is a numeric variable or expression.

Compatibility: Version 18 and higher

### **D\_Multiplier**

Declaration: Alias D\_MULTIPLIER &H0A1D

Arguments: One floating-point value representing the dimension's units multiplier(only active if D\_UNITS is equal to -1(no units))

Accessibility: Query/Change

Purpose: This parameter allows you to determine dimension's units multiplier.

Query Usage: Query D\_Multiplier, dmult where dmult is a numeric variable.

Change Usage: Change D\_Multiplier, dunits where dmult is a numeric variable or expression.

Compatibility: Version 18 and higher

### **D\_TextOffset**

Declaration: Alias D\_TEXTOFFSET &H0A1E

Arguments: One floating-point value representing the distance between the arrow tails and the dimension text.

Accessibility: Query/Change

Purpose: This parameter allows you to specify how much space is between the arrow tails and the dimension text. This distance has an effect on when the text and/or arrows automatically move outside the extension lines.

Query Usage: Query D\_TextOffset, doffset where doffset is a numeric variable.

Change Usage: Change D\_TextOffset, doffset where doffset is a numeric variable or expression.

Compatibility: Version 18 and higher

### **D\_ForceText\_Hor**

Declaration: Alias D\_FORCETEXT\_HOR &H0A1F

Arguments: A number (integer [0,1]) indicating whether the text should always be drawn horizontally or aligned with the dimension arrows. Values: 0 = no enforced alignment, 1 = text always drawn horizontally

Accessibility: Query/Change

Purpose: This parameter allows you to force text to always be drawn horizontal, or to always align with the dimension arrowheads.

Query Usage: Query D\_ForceText\_Hor, ihor where ihor is a numeric variable.

Change Usage: Change D\_ForceText\_Hor, ihor where ihor is a numeric variable or expression.

Compatibility: Version 18 and higher

## New Dimension Command Parameters:

The following Dimension commands have new command parameters:

- >DimAngle
- >DimArc
- >DimBase
- >DimChamfer
- >DimCoordinate
- >DimDiameter
- >Dimension
- >DimensionDistanceOnly
- >DimExtend
- >DimProgress
- >DimRadius
- >DimRadProgress

The new parameters are:

### < Associated a

[0 = not associated, 1 = associated (by segment for linear dimensions), 2 = associated (by entity, for linear dimensions only)]

### < EnableSecondArrow esa

[0 = second arrow type is disabled (first arrow type for both heads; 1 = different arrowheads)]

### <Arrowhead2 a

Second arrowhead style; accepts the same values as the <Arrowhead argument. Only affects the dimension if <EnableSecondArrow = 1



#### <DimUnits du

[-1 = no Units(multiplier is used); 0 = Unitless; 1 = Inches; 2 = Feet; 3 = Mile; 4 = Millimeters; 5 = Centimeters; 6 = Meters; 7 = Kilometers]

This parameter uses preset multiplier to convert the dimension distance from the drawing's base units to the selected units.

#### <DimEnableTextInsidePos tip

Optionally forces text to be drawn inside the dimension lines: [0 = free, 1 = inside only]

#### <DimHorizontalTextPos htp

[0=centered, 1=left, 2=right, 3=above left extension, 4=above right extension]

#### <DimVerticalTextPos vtp

[0=centered, 1=above arrows, 2=below arrows]

#### <Style dst

[0=normal, 1=bold, 2=italic, 3=bold and italic]

#### <Multiplier mult

If DimUnits = -1, the distance between the measurement points is multiplied by this value to generate the displayed distance.

### Changed Dimension Command Parameters:

The following dimension parameters have new behavior:

#### <Orientation ori

text orientation [0 = normal, 1 = flipped]

#### <Location dloc

[0 = centered, 1 = above, 2 = below, 3 = right, 4 = upper right, 5 = lower right, 6 = left, 7 = upper left, 8 = lower left.]

*Note: I'm sorry, but I can't yet say what interaction there may be if you specify both Location and DimVerticalTextPos/DimHorizontalTextPos in the same dimension command.*

## New Macro Commands

### >SuppressErrorDialog

```
{  
<Silent s [1=Suppress errors, 0=Display errors]  
}
```

Suppress DesignCAD's error message dialog.

Compatibility: version 18 and later

### >AddCustomMenu

```
{  
<Filename "filename.dmf"  
}
```

Note: As with the DxfIn command, you may use an asterisk in front of the filename for a shortcut to the DesignCAD directory, such as "\*\mymenu.dmf".

Adds custom menu (\*.dma, \*.dmf) files the current menu structure (convenient for loading your own menu category without affecting the main menu).

Compatibility: version 18 and later

Example:

```
>AddCustomMenu  
{  
<Filename "*\MyCustomMenu.dma"  
}
```

Example of \*.dma file (Note that the numbers must be added to the Dc3MaxKey.ini file and associated with the corresponding macros or other items.):

```
DesignCAD Popup Menu  
;Host Menu Item  
Plugin  
;PopupMenu to add  
POPUP "My Add-Ins"  
BEGIN  
MENUITEM "My First Macro", 25501  
MENUITEM "My Second Macro",25502  
MENUITEM "My Third Macro",25503  
MENUITEM SEPARATOR  
MENUITEM "Another Macro",25504  
MENUITEM SEPARATOR  
MENUITEM "Utility1",25505  
MENUITEM "Utility2",25506  
END
```

Example of Dc3MaxKey.ini file (the '\*' at the beginning of the path means the file is located under the DesignCAD home directory, typically "My Documents\DesignCAD 3D Max 18"):

```
[Menu-Command-File]
25501=*\Plugins\My Addins\MyMacro1.d3m
25502=*\Plugins\My Addins\MyMacro2.d3m
25503=*\Plugins\My Addins\MyMacro3.d3m
25504=*\Plugins\My Addins\AnotherMacro.d3m
25505=*\Plugins\My Addins\Util1.d3m
25506=*\Plugins\My Addins\Util2.d3m
```

## BasicCAD Changes in Version 18.1

### Support added for additional parameters in several commands.

#### >DxfOut

now supports the <SelectOnly parameter

#### >DwgOut

Now supports the <SelectOnly parameter

#### >SaveCopy

Now supports the <SelectOnly parameter

#### >DimRadius

Now supports the <Associated 0 parameter

#### >DimDiameter

Now supports the <Associated 0 parameter

## BasicCAD Changes in Version 18.2

### New Sys Function for DWG/DXF import settings

#### Sys(2012)

This Sys() function gets or sets the Varying-Width Entities radio button setting in Import Setup:  
[0=import as hatch fills, 1=import as single-width entity]

### New Import Command Parameters

Both >DwgIn and >DxfIn now support the following parameter:

#### <ImportVaryingWidthAsSingleWidth [iv]

[0=as hatch fills, 1=as single-width lines]

This parameter sets the Varying-Width Entities radio button in Import Setup.

All >Dimension commands now support the following new parameters:

#### **<OptDimTextOffset [txtoffset]**

This parameter specifies the size of the text offset.

#### **<Offset [baseoffset]**

This parameter is not new, but reverts to its original meaning as the baseline dimension offset.

#### **<ForceTextHorizontallyOutside [fto]**

This parameter sets the matching checkbox in the Dimension Text dialog: [0=unchecked, 1=checked].

<Text Movement [0=dim line with text, 1 = Add leader, 2 = no leader]

## **New Query/Change Parameters for Dimension Entities**

### **D\_ForceText\_Hor\_Out**

Declaration: Alias D\_FORCETEXT\_HOR\_OUT &H0A20

Arguments: A number (integer [0,1]) indicating whether the text should always be drawn horizontally when drawn outside the dimension arrows. Values: 0 = no enforced alignment, 1 = text always drawn horizontally if outside the arrows.

Accessibility: Query/Change

Purpose: This parameter allows you to force text to always be drawn horizontal if drawn outside the dimension arrows.

Query Usage: Query D\_ForceText\_Hor\_Out, ihout where ihout is a numeric variable.

Change Usage: Change D\_ForceText\_Hor\_Out, ihout where ihout is a numeric variable or expression.

Compatibility: Version 18.2 and higher

### **D\_TextMovement**

Declaration: Alias D\_TEXTMOVEMENT &H0A21

Arguments: A number (integer [0,1,2]) indicating how the text is placed relative to the arrows. Values: 0 = text stays in line with the arrows (subject to horizontal and vertical placement), 1 = text requires an extra point to be set indicating its placement, with a leader connecting it to the dimension, 2 = text requires an extra point to be set for placement, but with no leader connecting it to the dimension.

Accessibility: Query/Change

Purpose: This parameter allows you to specify the Text Movement setting for a dimension.

Query Usage: Query D\_Movement, mv where mv is a numeric variable.

Change Usage: Change D\_Movement, mv where mv is a numeric variable or expression.

Compatibility: Version 18.2 and higher

## BasicCAD Changes in Version 19.0

### New Sys Functions

#### Sys(1140)

This Sys function controls whether DesignCAD will or won't use nested groups. [ 0=nested groups, 1 = "old-style" groups]

#### Sys(1141)

This Sys function (Read-only) indicates the highest used group index.

#### Sys(1142)

This Sys function (Read-Only) indicates the lowest unused group index.

### New Command Parameters

The >Arrow command now supports the following additional parameter:

#### <CurvedArrow [ca]

This parameter indicates whether an arrow should be drawn straight or curved. [0=straight, 1=curved]

The >DwgIn and >DxfIn commands now support this additional parameter:

#### <SkipLayouts [sl]

This parameter indicates whether the import command should only import model-space information or also include paperspace layouts. [0=import layouts, 1 = skip layouts]

The >DimExtended command now properly supports the following parameter:

#### <Type [t]

This parameter is used to indicate whether or not to draw dimensions exploded. [0 = as dimension, 1 = exploded]

## BasicCAD Changes in Version 20.0

### New Sys() Functions

#### Sys(1150)

This Sys() function allows you to toggle the "Show/Hide Command Dialog" setting. [0 = show it, 1 = hide it]

## Sys(1151)

This Sys() function controls the automatic toggle of by-layer settings when you layers. [0 = toggle, 1 = don't toggle]

## BasicCAD Changes in Version 21.0

### New Sys Function

#### Sys(1160)

This Sys() function controls whether or not to use the new style of shadows when shading the drawing. [0=don't use, 1=use new shadows]. The new shadow functionality improves the appearance of shadows when shading a zoomed-in area of the drawing (less "jaggies" than the older algorithm).

## BasicCAD Changes in Version 22.0

### New Command Parameter

The >Sweep command now supports this new parameter:

#### <KeepOriginalProfile [kp]

This parameter matches (and sets) the new "Keep Original Profile" checkbox in the Sweep command. [0=don't keep, 1=keep]

New BasicCAD Commands

#### >ConvertDoubleLineToVector

This command is used to convert any selected double lines to vector outlines. The line style of the converted lines depends on the current line style or (if Line Style by Layer is active) the layer line style. It requires no parameters.

#### >LayerOptions2

This command is used to open the new Layer Options dialog. It requires no parameters.

## BasicCAD Changes in Version 22.1

### New Query/Change Parameters

#### AW\_CURVED

Alias AW\_CURVED &H0905 'R/W - b (0: false, 1: true) version 22.1 – indicates/sets the ability to draw an arrow with a curved body.

#### D\_EXTENSIONLENGTHFIXED

Alias D\_EXTENSIONLENGTHFIXED &H0A22 'R/W - b (0: variable extension length, 1: extension length fixed) version 22.1 – indicates/sets whether a dimension's extension line uses a fixed length or variable length.

#### D\_EXTENSIONGAP

Alias D\_EXTENSIONGAP\_LEN &H0A23 'R/W - f (sets/returns extension line length if length is fixed, otherwise sets/returns extension line gap size) version 22.1

#### D\_EXTENSIONOVERSHOOT

Alias D\_EXTENSIONOVERSHOOT &H0A24 'R/W - f (sets/returns the length of the extension line overshoot) version 22.1

#### D\_APPENDUNITSSUFFIX

Alias D\_APPENDUNITSSUFFIX &H0A25 'R/W - b (0: units suffix not appended, 1: units suffix auto-appended) version 22.1

#### SY\_ORIGINAL\_COLOR

Alias SY\_ORIGINAL\_COLOR &H0E0F 'R/W – b (0: don't use original color; 1: use original color) version 22.1 – implements BasicCAD support for the "use original color" option for blocks in the Info Box.

## BasicCAD changes in Version 23.0

### New Sys() Functions

#### Sys(171)

Print Black and White [0 = no, 1 = yes]

#### Sys(172)

Print Shadows [0 = no, 1 = yes]

#### Sys(490)

Read-only – acquires the index of an entity whose longid (64-bit id) was written to Sys\$(490)

### Sys(1143)

Read-only – Last-used Solid ID

### Sys(1144)

Read-only – Lowest unused Solid ID

### Sys(1173)

Display CS [0 = no, 1 = yes]

### Sys(1174)

Automatically pack Group IDs and Solid IDs on Save. [0 = no, 1 = yes]. This setting remains sticky between sessions.

**IMPORTANT NOTE:** Sys(1174), when set to 1, is currently conflicting with some save operations within BasicCAD. We suggest for now that if your macro is going to save a drawing, your macro code should set Sys(1174) to zero and use the >PackGroupIDs and >PackSolidIDs commands (if so desired) before saving a file.

## New Sys\$() Functions

### Sys\$(295)

Read-only – Retrieves the long id (permanent 64-bit id used by OLE Automation) of an entity selected by the Entity Statement as a string. An entity's index may change if other drawing entities are deleted, but so long as it remains unedited (no trimming or rearranging points) its 64-bit id should remain constant. This may prove handy for keeping long-term track of entities over the course of many drawing sessions.

### Sys\$(240)

Write-only – Store the string representation of an entity's 64-bit id here, then read Sys(490) to get that entity's index. If the 64-bit ID is not found, Sys(490) will be set to zero.

## New Query/Change Parameter

### ENT\_LONGID

Alias ENT\_LONGID &H0x011B 'readonly string representing the long id of an entity selected by the Entity statement.

## >DwgOut and >DxfOut Command Parameter Update

### <Type

The supported values to specify the target AutoCAD version are [7=2013, 6=2010, 5=2007, 4 = 2004, 3 = 2000, 2 = R14, 1 = R13, 0=R12]



## BasicCAD Changes in Version 24.0

### New Sys() functions:

#### Sys(1145)

enables (1) or disables (0) the "Use surface representation for solid operations" checkbox in General Options.

#### Sys(214), Sys(215), Sys(216)

Read-only floating-point values to help you determine the X, Y, and Z location of the camera's current target location – i.e. "what coordinate is the camera pointing at?" To change the camera target, run the SetViewerPoints command – you only need to set the first point, the second ("where's the camera?") is optional.

### New drawing commands:

#### >MatchProperties

No parameters are available yet; this just starts the Match Properties Tool command from a BasicCAD script. You'll still have to manually pick the "source" object and select parameters to be copied.

#### >SolidToSurface

This command converts the selected solids to the new Solid Surface entity type. No parameters.

#### >SurfaceToSolid

This command converts the selected Solid Surface entities to regular solids (note: The result will be all planes, no grids). No parameters.