

README

MapInfo Professional, version 5.5 BETA

Buffer Enhancements

Buffer functionality uses a *Width* setting to create a Buffer that is some measured distance from the outline of the object. Using Latitude/Longitude data, the perfect buffer width, as defined in native Latitude/Longitude decimal degrees, may change on different portions of an object. On small objects, this difference from node to node may not be measurable. On objects that span large distances, such as the United States, the distance may be measurable. MapInfo Professional calculates a native decimal degree width for one location in the object, typically the center of the bounding box. Thus, the measured On Earth (Spherical) distance from the boundary of the original input object to the boundary of the new buffered object may vary slightly from node to node.

In versions prior to MapInfo Professional 5.5, all Buffer calculations, except those from NonEarth Projections, were performed On Earth using the Spherical method described above.

In MapInfo Professional 5.5, you can produce Cartesian calculated buffers. Using this option, the data is considered to be in a flat-projected coordinate system, and the measured buffer widths are calculated using Cartesian distances. This produces exact buffers (as measured by the Cartesian Distance functions) as long as the data in **not** in a Latitude/Longitude projection.

To choose a distance and area calculation when buffering:

1. Choose Objects > Buffer.
2. In the Buffer Width Distance using:

Spherical

Spherical calculations are used for distance methods, which attempt to keep the measurement on the curved surface of the earth. The data is first converted to Latitude/Longitude and then a calculation is produced. Non-Earth projections cannot use spherical calculations.

Cartesian

Cartesian methods are used to perform calculations on data projected onto a flat plane. Cartesian coordinates are a pair of numbers, (x, y) , defining the position of a point in a two-dimensional space by its perpendicular projection onto two axes which are at right angles to each other. Long/Lat projections cannot use Cartesian calculations.

Default

The method used in MapInfo Professional pre-5.5 versions for performing distance calculations for polyline length and region perimeter is a MapInfo method called the On Earth Straight Line approximation method. In this calculation, the data is converted to Latitude/Longitude, and then a straight line distance approximation is used so that the calculation does not take into account the curvature of the Earth. Measuring distance between two points, including area calculations using the Ruler Tool uses the Spherical method.

MapInfo Professional default calculation method is the On Earth Straight Line approximation method. In this calculation, the data is converted to Latitude/Longitude; a straight line distance approximation is used so that the calculation does not take into account the curvature of the Earth. These calculations include Polyline Length and Region Perimeter.

Only valid choices are available for the target table's projection. Thus, if the target table is in a Latitude/Longitude projection, then Cartesian is not a valid option, and the Cartesian option is not available. If the target table's projection is NonEarth, Great Circle is not a valid option, and the Great Circle option is not available.

Concentric Ring Buffer Tool

Access the enhanced Concentric Ring Buffer tool by selecting Tools > Tool Manager. Select Concenmtric Ring Buffers. Check Load and Autoload. The tool will be added to the Tools menu.

Use the Concentric Ring Buffer tool to create multiple concentric ring buffers around one or more objects.

Buffer Radii: Displays the buffer radii as designated in the following dialog prompts.

Buffer Style:

Fill Style: Choose a fill style for the buffer ring.

Line Style: Choose a line style for the outline of the buffer ring.

Smoothness: Specify the segments per circle to determine the resolution of the curves in the Buffer polygon. More segments produce smoother curve; fewer segments make a more jagged curve.

Enter New Buffer Ring Here (smallest to largest):

Radius: Specify the buffer radius, the distance between the object or objects and the border of the buffer polygon. Enter the number of units for the ring buffer.

Units: Select a unit from the drop-down list.

Add Ring: Select Add Ring to add the buffer to the Buffer Radii list.

Change Ring: Select radii from the list to edit it.

Delete Ring: Select radii from the list to delete it.

Name of Table to Store the Buffers In: Assign a table name and directory.

Choose OK to create the buffer(s); choose Clear to delete all information from the Buffer Radii box.

Create Legend Enhancements

The Create Legend process now allows you to create attribute driven legends. This makes it easier to create legends because text

descriptions can be automatically generated. Legends can be created based on unique map styles, or by unique values in an attribute column. Text descriptions can be generated from attribute columns and/or expressions. Additionally, the process by which the legend was generated can be saved to the map table metadata as a default that will be invoked when the legend is next created. The metadata defaults are used to initialize the Create Legend Step 3 dialog. Override metadata defaults and select your own settings in the dialog and save your changes by writing them to the map table metadata.

Create Legend Step 3 Dialog

Choose Next at the Create Legend Step 2 of 3 dialog to display the Create Legend Step 3 of 3 dialog.

Select and set attributes for each Legend Frame.

Legend Frames:

The list box includes all frames selected in Step 1 (including all theme frames). The list is single select. Select one frame and set attributes for it (or accept the defaults). Each frame in the list represents a layer (and underlying table) in the map.

Title: Designate a title for the selected legend frame.

Subtitle: Designate a subtitle for the selected legend frame.

The title and subtitle are either from the defaults indicated in the Step 2 of 3 Create Legend dialog or from the metadata in the .tab file. If metadata keys do not exist, the Legend Frame Defaults from Step 2 are used. See the following section: Save frame Settings to Metadata.

Styles from: unique map styles:

unique values in selected column: Select from a list of columns in the table

Label styles with: list of columns in the table + <Style Name Pattern> + Expression...

Creating a Customized Label

To create a customized label, choose Expression... to display the Expression dialog box. In the Expression dialog box you can create multiline labels (using CHR\$(10)) that references several columns of text.

Put single quotation marks around text in label expressions.

Working with Theme Legends

Only title/subtitles can be set for theme frames. If a theme layer is selected, the title/subtitle currently set for the thematic legend displays; you may edit these. The other options are not available. Changing a theme legend's title/subtitle affects all occurrences of the thematic legend. Additionally, fonts designated in Step 2 will not be applied to thematic legends.

Display Style Overrides Are in Effect

When the selected layer has style overrides active, the check box for Display style overrides is checked and the column and label fields are ignored when the legend is created. However, if Display style overrides is not active for the map layer and the legend is refreshed, it will be refreshed according to the column and label originally designated. The check box indicates the status of styles overrides; it cannot be checked or unchecked.

Save Frame Settings to Metadata

Choose 'Save frame settings to Metadata' to save the settings for Title, Subtitle, styles column and styles label to the table metadata for the selected legend frame. If you select this option, the designated settings are saved to the table metadata, and become the defaults for the legend frame.

Metadata Support For Step 3

If you have saved the frame settings to the table metadata, the map layer table is updated accordingly.

The metadata keys are:

```
"\legend" = ""
"\legend\title" = "a string value"
"\legend\subtitle" = "a string value"
"\legend\column" = "a column identifier"
"\legend\label" = "an expression"
```

Example:

```
"\legend" = " "
"\legend\title" = "Legend"
"\legend\subtitle" = " "
"\legend\column" = "type"
"\legend\label" = "Ucase$(type)"
```

If there is no metadata for \legend\column, the default in the UI and MapBasic is:

Styles from = unique map styles (e.g., the object column)

If there is no metadata for \legend\label, the default in the User Interface and MapBasic is:

Label styles with = <Style Name Pattern>

Therefore, the current value of Style Name Pattern from Step 2 is used to generate legend text, based upon Legend Preferences. If there is no metadata for \legend\title or \legend\subtitle, the default will be based on the Title and Subtitle patterns defined in Legend Preferences.

Metadata Keys and Style Override

We support the following metadata keys when Display Style override is active. If these keys exist, their value is used for the legend text.

```
\legend\allpoints = some string value
\legend\alllines = some string value
\legend\allregions = some string value
```

Creating Metadata Keys

Embedded string literals in label expressions have to be double quoted to be supported in metadata. For example, to add the string constant "Feature Code" to an expression that returns the uppercase string from the column fcode, set the metadata key/value like this:

```
"\legend\label" = "Ucase$(fcode) + "" Feature Code """
```

Note that a metadata key such as:

```
"\legend\label" = ""
```

does not mean that the label expression is an empty string. It means that there is NO metadata available for the "\legend\label" key, so the default will be <Style Name Pattern>.

Extended Metadata Support (pass-through metadata keys)

For data that has feature codes, but no feature description, there is additional metadata support, called Pass-through Keys. The keys map column values to descriptive text used in the legend. Table metadata contains 'pass-through' text for legends if the following criteria are met:

1) the metadata label expression is a simple column reference.

2) a pass-through prefix key exists in the metadata that is the result of appending the column reference to the top level "\legend" key, for example, given the following legend label expression key:

```
"\legend\label" = "CFCC"
```

the following pass-through prefix key must also exist in metadata:

```
"\legend\CFCC"
```

The prefix key may exist as a separate line in metadata:

```
"\legend\CFCC" = ""
```

If the above two criteria are met, then the legend creation code looks for legend text in additional 'pass-through' metadata keys that are the result of appending evaluated label expressions to the prefix key. The values for each key replace the original expression value:

```
"\legend\CFCC\F32" = "Church"  
"\legend\CFCC\F34" = "Hospital"  
.  
.  
"\legend\CFCC\value_n" = "legend text used in place of CFCC value_n"
```

This results in a legend with the descriptive text "Church", "Hospital" etc., rather than "F32", or "F34".

Metadata Pass-through Key Limitations

Once a Legend Frame that uses pass-through metadata keys for text is created, changes to the pass-through key values do not display in a Legend Refresh. They will display the next time the Legend Frame is recreated.

Date Window for 2-Digit Year

MapInfo Professional now includes a date window that converts two-digit input into four digit years. It also allows you to change the default to one that best suits your data.

The ability to change the date window can be accessed as a:

- setting in the Options > Preferences > Systems Settings dialog.
- MapBasic command.

Date Window for 2-Digit Years Option Using the System Settings Dialog

To access the 2-Digit Years option:

1. Choose Options > Preferences > Systems Settings
 - Turn date windowing off (use current century): Use the current century; if you choose this option the Set date window to option is disabled.
 - Set date window to: Enter a number from 0-99. The number you enter displays in the statements below the prompt that indicate whether the date will display with the prefix 19 or 20.
For example if you enter the number 50, the statements will indicate that:
Years entered as 00-50 become 2000-2050.

Years entered as 51-99 become 1951-1999.

Displaying Coordinates in Degrees, Minutes, Seconds

MapInfo Pro 5.50 allows for the displaying of coordinates in degrees, minutes, seconds (e.g., 75 degrees 12' 48") in addition to decimal degrees (e.g., 75.213458 degrees) wherever coordinates can be displayed.

Choose to display coordinates in degrees, minutes, seconds by accessing the Map Window Preferences dialog or the Map Options dialog.

Setting the Preferences for Coordinate Display Using the Map Window Preferences Dialog

Access the Map Window Preferences dialog box (Options > Preferences > MapWindow) to specify how MapInfo displays the coordinates. When the "Show Degrees in Degrees Minutes Seconds" checkbox is active, coordinates display be displayed in degrees, minutes, and seconds; when unchecked, coordinates will be displayed in degrees decimal. The default is degrees in decimal.

Overriding the Map Window Preferences Dialog Settings

Access the Map Options dialog when a map window is active. When the "Show DMS" checkbox is active, coordinates display in degrees, minutes, and seconds; when unchecked, coordinates display in degrees decimal. The default is degrees in decimal.

Coordinate Display

MapInfo displays coordinates:

- In the status bar by cursor location (designated in the Map Options dialog or by clicking on the status bar)
- In various Object Info dialogs (e.g., Point Object, Region Object, etc.)

Enhanced Ruler Window Functionality Display Distance While Drawing Objects

The functionality of the ruler window has been expanded. If the ruler window is active, the ruler tool will show the distance when drawing or selection tools are being used.

The Ruler window shows the current segment distance, as well as a cumulative distance. If necessary, a rubber banding line displays to show what points are being used to compute the distance. The type of object being drawn or used for selection determines whether the rubber-banding line displays.

Tools that display an additional rubber-banding line:

- Arc drawing tool
- Ellipse drawing tool
- Rectangle drawing tool
- Round Rectangle drawing tool
- Marquee Select tool

Radius Select tool

If the ruler window is not active, the drawing and selection tools will not display distance and the additional rubber banding line will not be visible.

Enhancements to the Distance and Area Calculation Methods

The method used in MapInfo Professional pre-5.5 versions for performing distance calculations for polyline length and region perimeter is a MapInfo method called the On Earth Straight Line approximation method. In this calculation, the data is converted to Latitude/Longitude, and then a straight line distance approximation is used so that the calculation does not take into account the curvature of the Earth. Measuring distance between two points, including using the Ruler Tool and area calculations used the Spherical method. The Spherical method continues to be the calculation method for area calculations in MapInfo Professional 5.5 unless the data is in Non-Earth projections, then Cartesian is used.

MapInfo Professional 5.5 includes the option to perform calculations using the Spherical and Cartesian methods. Set the Distance and Area calculation method in the Map Window Preferences dialog or the Map Options dialog. The Map Options dialog only presents those options that are appropriate to the active map; therefore latitude/longitude maps cannot use the Cartesian method.

The Preference Settings determine what method is used for any new map created in MapInfo Professional 5.5; if you open a workspace created in pre 5.5 versions, the currently designated setting for area and distance calculation will be used.

Setting Distance and Area Calculations Using the Map Window Preferences Dialog

To set distance and area calculations using the Map Window Preferences Dialog:

1. Choose Options > Preferences > Map Window.
2. From the **Distance/Area using:** option, choose a calculation method:

Spherical

Spherical calculations are used for distance methods, which attempt to keep the measurement on the curved surface of the earth. The data is first converted to Latitude/Longitude and then a calculation is produced. Non-Earth projections cannot use spherical calculations.

Cartesian

Cartesian methods are used to perform calculations on data projected onto a flat plane. Cartesian coordinates are a pair of numbers, (x, y), defining the position of a point in a two-dimensional space by its perpendicular projection onto two axes which are at right angles to each other. Long/Lat projections cannot use Cartesian calculations.

Setting Distance and Area Calculations Using the Map Options Dialog

To Set Distance and Area Calculations Using the Map Options dialog:

1. Choose Map > Options.
2. From the **Distance/Area using:** option, choose a calculation method:

Spherical

Spherical calculations are used for distance methods, which attempt to keep the measurement on the curved surface of the earth. The data is first converted to Latitude/Longitude and then a calculation is produced. Non-Earth projections cannot use spherical calculations.

Cartesian

Cartesian methods are used to perform calculations on data projected onto a flat plane. Cartesian coordinates are a pair of numbers, (x, y), defining the position of a point in a two-dimensional space by its perpendicular projection onto two axes which are at right angles to each other. Long/Lat projections cannot use Cartesian calculations.

MapInfo Professional default calculation method is the On Earth Straight Line approximation method. In this calculation, the data is converted to Latitude/Longitude; a straight line distance approximation is used so that the calculation does not take into account the curvature of the Earth. These calculations include Polyline Length and Region Perimeter.

Only valid choices are available for the target table's projection. Thus, if the target table is in a Latitude/Longitude projection, then Cartesian is not a valid option, and the Cartesian option is not available. If the mapper's projection is NonEarth, Great Circle is not a valid option, and the Great Circle option is not available.

Identifying the Area and Distance Calculation Used

To identify the area and distance calculation used:

Double-click on an object type, including Lines, Rectangles, Rounded Rectangles, Region, Polyline, and Ellipses to display the Object information box about the object. The type of calculation method used to calculate measurement displays in parenthesis after the measurement. LL Straight is the MapInfo default method: On Earth Straight Line discussed previously.

Grid Theme Enhancements

Buttons in the Create Thematic Step 3 of 3 dialog provides access to the settings, styles, and legend dialogs. Choose the Styles button to display Grid Appearance dialog where grid dimension and inflection color/value are specified. Choose the Settings button to invoke a dialog where interpolator settings can be specified. The Legend button displays a dialog where legend settings are specified.

The Interpolator list box and Setting buttons are disabled when modifying an existing grid that has no interpolation settings. If these grids were created with MapInfo Professional, the settings are persisted in the grid's TAB file as metadata. If the grid was created with Vertical Mapper v1 or v2, or with MapInfo's DBM converter, interpolation settings are not available.

Create Thematic Map – Step 3 of 3

In the Create Thematic Step 3 of 3 dialog, choose Styles to display the Grid Appearance dialog.

Grid Appearance Dialog

Method: There are four different methods that determine where inflection values occur. Select the method for distributing inflections. Equal Cell count and Custom Cell Count are based on percentages. Equal Value Ranges and Custom Value Ranges are based on values.

Equal Cell Count

Sets the inflections so that approximately an equal number of grid cells falls between each inflection value. Inflection breaks can be expressed as a percentage of the whole. For example, in a legend for a grid whose 4 inflections have been spread by equal cell count, 33% of the grid cells will have colors between dark blue and light blue (the colors of the first two inflections), 33% of the grid cells will have colors between light blue and green, 33% of the grid cells will have colors that fall between green and red. Additionally, the percentages are additive. 67% of the grid cells will have colors between dark blue and green. And 100% or all of the grid cells will have colors between dark blue and red.

When the grid cells are interpolated, these percentages can be converted to actual values. If the inflection spread is based on cell count (equal or custom), you will see percentages in the inflection listview and in the preview legends. You won't see the actual values until the grid is actually computed.

When the spread method is based on cell count, anytime you change an interpolator parameter or choose to use a different interpolator for an existing grid theme, you will see percentages in the preview legend.

Changing the number of inflection values in the combo box will respread the cell count percentages. So, if the number of inflections changes from 4 (with equal cell count having set the percentages to 0%, 33%, 67%, and 100%) to 6, the percentages would be 0%, 20%, 40%, 60%, 80%, and 100%.

Equal Value Ranges

The equal range method spreads the inflections evenly between the minimum and maximum of the source data range.

Custom Cell Count

Choose this method when you want to use your own percentages and not ones that are evenly spread.

Custom Value Ranges

Choose this method when you want to use your own values and not ones that are evenly spread.

Number of Inflections: Select from a list of 2 to 16 inflections or type a number between 2 and 255.

Round By: Choose from a list of rounding factors to be applied to inflection values. You may not see the effects of this rounding if the spread method is based on cell count until inflection values are actually calculated.

Inflection Color/Value List: Control color, percentage/value inflection pairs. Items in the list are sorted by percentage/value. Edit the inflection percentages/values by single or double clicking on them with the left mouse button; you can edit a value by typing over it. Double click on a color swatch to display a color picker and change the color. If you edit a percentage while the method is equal cell count, the method becomes custom cell count. If you edit a value while the method is equal value ranges, the method becomes custom value ranges. The vertical bar to the right of the listview control shows the spectrum of colors used in the grid.

Default Colors for New Inflections

When you increase the number of inflections, MapInfo Professional chooses default colors for those new inflections. It keeps the last inflection color the same but inserts new colors that fall between the new last color and the one before it.

Color Adjustments

Contrast: Slide the bar between 0 and 100% to set the grid's contrast level.

Brightness: Slide the bar between 0 and 100% to set the grid's brightness level.

Gray Scale: Check to show only gray scale colors in the grid.

Flip Colors: Click this button to invert the inflection colors.

Relief Shading: Shaded relief maps are grid maps whose grid cell colors have had their brightness adjusted to indicate surface slope and slope direction relative to a user-defined light source direction. MapInfo Professional determines the orientation of each cell in the grid, and assigns a brightness to each grid cell.

Brightness (or reflectance) values correspond to light striking the surface. The light source can be thought of as the sun shining on a topographic surface. The maximum brightness is assigned at points where the rays are perpendicular to the surface (where the slope directly faces the light source). As slope faces turn away from the light source, lower reflectance values are assigned.

Light Source:

The Light Source Options let you specify the orientation of the light source. Check to activate relief shading and/or generate relief shading information in the grid.

Horizontal Angle: Rotate the light source in the horizontal plane. Zero degrees corresponds to the light source shining from due East. Positive angles rotate the light source counterclockwise, so, for example, 90 degrees places the light source due North.

Vertical Angle: Rotate the light source in the vertical plane. Zero degrees places the light source at the horizon, and 90 degrees places it directly overhead. Specify angles between 180 and 360 degrees to have the light source originate from under the surface.

Vertical Scale Factor: Specify a scale factor for the z values. Increasing this scale factor exaggerates the surface vertically, enhancing the shading effect. This can be useful for enhancing detail in relatively flat surfaces.

Inflection Values

The minimum and maximum values in the source table are shown here. These are not editable.

Displaying the IDW Settings Dialog

Choose Settings in the Create Thematic Map Step 3 of 3 dialog to display the IDW Settings Dialog.

Cell Size: Define the width and height of a grid cell in distance units. Since Grid cells are square both width and height are specified with one value. When you change the cell size, the Grid Dimensions static text control updates to reflect the new dimensions.

Grid Dimensions: Static text control displaying grid dimensions in cells.

Exponent: Defines the exponential influence of nearby data points on the value calculated for each grid cell. Increasing the exponent *decreases* the influence of data points the further they are from a grid cell.

Search Radius: Defines the maximum distance in grid cells between a grid cell and its neighboring data points. The distance determines if these points are considered in the distance weighing average.

Grid Border: Defines the number of cells beyond the source table's bounding rectangle by which to expand the grid.

Aggregate Coincident Points by: Choose how to aggregate the z-values of source data points that occupy the same grid cell. The values can be counted, averaged, summed, or the minimum or maximum can be taken.

MrSID Raster Handler

MapInfo 5.5 includes the MrSID raster handler. This allows you to open and display raster images compressed in the MrSID format. Install the MrSID raster handler, in the same directory as MapInfo Professional, during the MapInfo 5.5 installation process. Choose the Install MrSid button to install MrSid. A sample .sid file, a raster image of Paris, is included.

MrSID format image handler software is licensed from Schlosser Geographic Systems, Inc.

To open a .sid file:

1. Choose File > Open Table. The Open Table dialog displays.
2. Select the directory in which the .sid files are located.
3. In the Files of type drop-down list, choose Raster Image.
4. Select the .sid file you want to open and choose Open or double-click on the file name.
5. The file displays; register it as you register any raster file.

Enhanced Grapher

You can now create new graph types, including 3-D graphs using MapInfo Professional's 5.5 3-D graph feature. Graph types include: area graphs, bubble graphs, histogram, 3D Graphs, pie rings, scatter charts.

To use the new graph feature:

1. Open a table containing a numeric field: integer, small integer, float, decimal.
2. Choose Window > New Graph. The Create Graph Step I of 2 dialog displays.
3. Choose a graph type and template. Choose Next.
4. The Create Graph Step 2 of 2 dialog displays.
 - Table: Choose a table from the drop-down list.
 - Field: Choose a field or fields.
 - Label with column: Choose a field from the drop-down list that will be used as the label.

Series in column.
Series in Row.
The graph displays with the selections you have made.

To edit the new graph:

Right click on the graph.
OR
Choose Graph > Menu
To display the menu options.

If the graph is a 3-D graph, the 3-D view Angle option is enabled.

The options displayed in the General Options; Series Options; Grids and Scale and Titles and Label menus vary according to the type of graph.

Formatting: Select an object in the graph and assign attributes to the object.

Save as Template: Save graph settings; the template is assigned a .3tf extension and can be used only with the graph feature.

Main Toolbar-- New Tools

There are two new tools on the Main Toolbar:

Polygon Selection Tool: Use this tool to perform a selection within a polygonal area.

Unselect All Tool: This tool enabled when a selection is made. Press the button to unselect all selections. To perform the same action from the menu, select Query > Unselect All.

ODBC Drivers

MapInfo Professional 5.5 supports these drivers:

SpatialWare 3.5
SpatialWare 3.5 (IUS)
SpatialWare 4.0
DB2/2 (obtainable from the WEB)
MS Access

The following supported drivers are the renamed Intersolv drivers:

Oracle 7
Oracle 8
SQL Server
Sybase 11
Informix 9 (IUS)

Scale Bar Tool

Access the new ScaleBar tool by selecting Tools > Tool Manager. Check Load and Autoload. The tool will be added to the Tools menu.

Width of scale bar: Enter a number and select a measurement unit for the scale bar.

Adjust distance between ends to account for curvature during projection. Check to adjust the scale to accommodate the curvature of the projection.

Font for scale bar labels: Select a font for the scale bar label.

Fill color for scale bar: Select a fill color the scale bar label.

Choose OK to draw the scale bar.

Set Coordinate System Bounds Tool

The Set Coordinate System Bounds Tool has two new features:

- Ability to automatically create a rectangle at the bounds.
- Ability to copy Bounds clause from the dialog for use in the MapBasic window.

Snap to Node Enhancements

Each window now has its own snap mode state; for example, if three windows are open, each window can have snap mode activated/not activated.

Toggle snap mode by pressing the 's' key. If snap mode is active for a window, "snap" displays in the status bar.

Additionally, the snap mode setting persists in workspaces.

MapBasic Enhancements:

Add Cartographic Frame statement

A new Using clause sets the legend column and text expression for each Frame.

The Using clause will increase the workspace version from 500 to 550.

If a Column is defined, it must be an attribute column name from the frame layer's table, or the object column. The legend styles are based on the unique styles in the mapfile. The default is object.

If a Label is defined, it can be defined as a valid expression or default. Meaning. The default frame style pattern is used when creating each style's text unless the style clause contains text. The default is default.

Initially, each frame layer's TAB file will be searched for metadata values for Title, Subtitle, Column and Label. If no metadata value exists for Column, the default is 'object'. If no metadata value exists for Label, the default is the default frame style pattern. If legend metadata keys exist and you want to override them, you must use the corresponding mapbasic syntax.

Example

Add Cartographic Frame

```
[ Window legend_window_id ]
[ Custom ]
[ Default Frame Title { def_frame_title } [ Font... ] ]
[ Default Frame Subtitle { def_frame_subtitle } [ Font... ] ]
[ Default Frame Style { def_frame_style } [ Font... ] ]
[ Default Frame Border Pen... ]
Frame From Layer { map_layer_id | map_layer_name }
  [ Using
    [ Column { column | object } ]
    [ Label { expression | default } ]
  ]
  [ Position ( x , y ) [ Units paper_units ] ]
  [ Title [ frame_title ] [ Font... ] ]
  [ SubTitle [ frame_subtitle ] [ Font... ] ]
  [ Border Pen... ]
  [ Style [Font... ] [ NoRefresh ]
    [ Text { style name } { Line Pen... | Region Pen... Brush... |
      Symbol Symbol... } ]
    [ , ... ]
  ]
[ , ... ]
```

Alter Object statement

The Centroid of a Region can now be set by using the Alter Object using the following syntax:

Alter Object *Obj* Geography OBJ_GEO_CENTROID, *PointObj*

Note that *PointObj* is a point object. This differs from other values input by Alter Object Geography, which are all scalars. A point is needed in this instance because we need two values which define a point. The Point that is input is checked to make sure it is a valid Centroid (i.e., it is inside the region). If the *Obj* is not a region or polyline, or if *PointObj* is not a point object, or if the point is not a valid centroid, then an error is returned.

An easy way to center an X and Y value for a centroid is as follows:

```
Alter Object Obj Geography OBJ_GEO_CENTROID, CreatePoint(X, Y)
```

The user can also query the centroid by using the ObjectGeography function as follows:

```
PointObj = ObjectGeography(Obj, OBJ_GEO_CENTROID)
```

Note that there are other ways to get the Centroid, including the Centroid(), CentroidX(), and CentroidY() functions. These functions remain unchanged.

OBJ_GEO_CENTROID is defined in mapbasic.def.

CartesianArea () function

New function. Returns the area using a cartesian algorithm.

Syntax

CartesianArea(*expr*, *unit_name*)

obj_expr is an object expression

unit_name is a string representing the name of a distance unit (e.g. "km")

Return Value

Float

Description

The **CartesianArea()** function will always return the area using a cartesian algorithm. A value of -1 will be returned for data that is in a Latitude/longitude coordinate system, since Latitude/Longitude data is not projected and not cartesian.

CartesianBuffer () function

New function. Returns the calculate the buffer by assuming the object is in a flat projection and using the *width* to calculate a cartesian distance calculated buffer around the object.

Syntax

CartesianBuffer (*inputobject*, *resolution*, *width*, *unit_name*)

inputobject is an object expression

resolution is a SmallInt value representing the number of nodes per circle at each corner

width is a Float value representing the radius of the buffer; if *width* is negative, and if *inputobject* is a closed object, the object returned represents an object smaller than the original object

unit_name is the name of the distance unit (e.g. "mi" for miles, "km" for kilometers) used by *width*

Return Value

Region Object.

Description

The **CartesianBuffer()** function will calculate the buffer by assuming the object is in a flat projection and using the *width* to calculate a cartesian distance calculated buffer around the object. If the *inputobject* is in a NonEarth Projection, then Cartesian calculations will be used regardless of the Buffer function called.

CartesianDistance () function

New function. Returns the distance between two locations using a cartesian algorithm.

Syntax

CartesianDistance (*x1* , *y1* , *x2* , *y2* , *unit_name*)

x1 and *x2* are x-coordinates (e.g. longitude)

y1 and *y2* are y-coordinates (e.g. latitude)

unit_name is a string representing the name of a distance unit (e.g. "km")

Return Value

Float

Description

The **CartesianDistance()** function will always return a value using a cartesian algorithm. A value of -1 will be returned for data that is in a Latitude/longitude coordinate system, since Latitude/Longitude data is not projected and not cartesian.

CartesianObjectLen () function

New function. Returns the area geographic length of a line or polyline object using a cartesian algorithm.

Syntax

CartesianObjectLen(*expr* , *unit_name*)

expr is an object expression

unit_name is a string representing the name of a distance unit (e.g. "mi" for miles)

Return Value

Float

Description

The **CartesianObjectLen()** function will always return a value using a cartesian algorithm. A value of -1 will be returned for data that is in a Latitude/longitude coordinate system, since Latitude/Longitude data is not projected and not cartesian.

CartesianPerimeter () function

New function. Returns the perimeter using a cartesian algorithm.

Syntax

CartesianPerimeter(*expr* , *unit_name*)

obj_expr is an object expression

unit_name is a string representing the name of a distance unit (e.g. "km")

Return Value

Float

Description

The **CartesianPerimeter()** function will always return a value using a cartesian algorithm. A value of -1 will be returned for data that is in a Latitude/longitude coordinate system, since Latitude/Longitude data is not projected and not cartesian.

Create Cartographic Legend statement

A new Using clause sets the legend column and text expression for each Frame.

The Using clause will increase the workspace version from 500 to 550.

If a Column is defined, it must be an attribute column name from the frame layer's table, or the object column. The legend styles are based on the unique styles in the mapfile. The default is object.

If a Label is defined, it can be defined as valid expression or default. Meaning. The default frame style pattern is used when creating each style's text unless the style clause contains text. The default is default.

Initially, each frame layer's TAB file will be searched for metadata values for Title, Subtitle, Column and Label. If no metadata value exists for Column, the default is 'object'. If no metadata value exists for Label, the default is the default frame style pattern. If legend metadata keys exist and you want to override them, you must use the corresponding mapbasic syntax.

Example

Create Cartographic Legend

```
[ From Window map_window_id ]
[ Position ( x , y ) [ Units paper_units ] ]
[ Width win_width [ Units paper_units ] ]
[ Height win_height [ Units paper_units ] ]
[ Window Title { legend_window_title } ]
[ ScrollBars { On | Off } ]
[ Portrait | Landscape | Custom ]
[ Default Frame Title { def_frame_title } [ Font... ] ]
[ Default Frame Subtitle { def_frame_subtitle } [ Font... ] ]
[ Default Frame Style { def_frame_style } [ Font... ] ]
[ Default Frame Border Pen... ]
Frame From Layer { map_layer_id | map_layer_name }
  [ Using
    [ Column { column | object } ]
    [ Label { expression | default } ]
  ]
  [ Position ( x , y ) [ Units paper_units ] ]
  [ Title { frame_title [ Font... ] } ]
  [ SubTitle { frame_subtitle [ Font... ] } ]
  [ Border Pen... ]
  [ Style [Font...] [ Norefresh ]
    [ Text { style_name } { Line Pen... | Region Pen... Brush... |
      Symbol Symbol... } ]
```

Create Grid statement

The filespec parameter has been changed and By Percent and Round have been used in the following syntax.

Create Grid

```
From tablename
With expression [ Ignore value_to_ignore ]
Into filespec [ Type grid_type ]
[ Coordsys ... ]
[ Clipping { Object obj } | { Table tablename } ]
Inflect num_inflections [ By Percent ] at
  color : inflection_value
  [ color : inflection_value ...]
[ Round rounding_factor ]
Cell { [ Size cell_size [ Units distance_unit ]] |[ Min num_cells ]}
[ Border num_cells ]
Interpolate With interpolator_name Version version_string Using
  num_parameters parameter_name : parameter_value
  [ parameter_name : parameter_value ... ]
```

filespec specifies the fully qualified path and name of the new grid file. In this version, the filespec must be a fully qualified pathname of the grid file including the extension.

CreateObject statement

A new Type sub-clause has been added.

Syntax

```
Create Object As { Buffer | Union | Intersect | Merge }
  From fromtable
  [ Into { Table intotable | Variable varname } ]
  [ Width bufferwidth [ Units unitname ] [Type keyword
spherical or cartesian] ]
  [ Resolution smoothness ]
  [ Data column = expression [ , column = expression . .
. ] ]
  [ Group By { column | RowID } ]
```

It can be MAPPER_INFO_DIST_SPHERICAL, or MAPPER_INFO_DIST_CARTESIAN .

The optional Type sub-clause lets you specify the type of distance calculation used to create the buffer. If the MAPPER_INFO_DIST_GREAT_CIRCLE type is used, then the calculation will be done by mapping the data into a Latitude/Longitude On Earth projection and using widths measured using Great Circle distance calculations. If the MAPPER_INFO_DIST_CARTESIAN type is used, then the calculation is done by considering the data to be projected to a flat surface and widths are measured using cartesian distance calculations. In the case of Buffer, MAPPER_INFO_DIST_DEFAULT is equivalent to MAPPER_INFO_DIST_GREAT_CIRCLE. If the Width clause does not include the Type sub-clause, then the default distance calculation type MAPPER_INFO_DIST_DEFAULT is used. If the data is in a

Latitude/Longitude Projection, then Great Circle calculations will be used regardless of the Type setting. If the data is in a NonEarth Projection, the Cartesian calculations will be used regardless of the Type setting.

DateWindow() function

A new function. Returns the current date window setting as an integer in the range 0 to 99, or (-1) if date windowing is off.

Syntax

DateWindow(*context*)

context A SmallInt that can either be DATE_WIN_CURPROG or DATE_WIN_SESSION.

Description

This depends on which context is passed. If *context* is DATE_WIN_SESSION, then the current session setting in effect is returned. If *context* is DATE_WIN_CURPROG, then the current MapBasic program's local setting is returned, if a program is not running the session setting is returned.

Spherical () function

New function. Returns the area using as calculated in a Latitude/Longitude non-projected coordinate system using great circle based algorithms.

Syntax

SphericalArea(*expr*, *unit_name*)

unit_name is a string representing the name of an area unit (e.g. "sq km")

Return Value

Float

Description

The **SphericalArea()** function will always return the area as calculated in a Latitude/Longitude non-projected coordinate system using great circle based algorithms. A value of -1 will be returned for data that is in a NonEarth coordinate system since this data can't be converted into a Latitude/longitude coordinate system.

SphericalBuffer () function

New function. Returns a region object that represents a buffer region (the area within a specified buffer distance of an existing object).

Syntax

SphericalBuffer (inputobject, resolution, width, unit_name)

inputobject is an object expression

resolution is a SmallInt value representing the number of nodes per circle at each corner

width is a Float value representing the radius of the buffer; if width is negative, and if inputobject is a closed object, the object returned represents an object smaller than the original object

unit_name is the name of the distance unit (e.g. "mi" for miles, "km" for kilometers) used by width

Return Value

Region Object

Description

The **SphericalBuffer**() function will calculate the buffer by converting the object into Latitude/Longitude On Earth projection and using the *width* to calculate a Great Circle distance calculated buffer around the object. If the *inputobject* is in a Latitude/Longitude Projection, then Great Circle calculations will be used regardless of the Buffer function used.

SphericalDistance () function

New function. Returns the distance between two locations.

Syntax

SphericalDistance (x1 , y1 , x2 , y2 , unit_name)

x1 and x2 are x-coordinates (e.g. longitude)

y1 and y2 are y-coordinates (e.g. latitude)

unit_name is a string representing the name of a distance unit (e.g. "km")

Return Value

Float

Description

The **SphericalDistance**() function always returns a value as calculated in a Latitude/Longitude non-projected coordinate system using great circle based algorithms.. A value of -1 will be returned for data that is in a NonEarth coordinate system since this data can't be converted into a Latitude/longitude coordinate system.

SphericalObjectLen () function

New function. Returns the geographic length of a line or polyline object.

Syntax

SphericalObjectLen(*expr* , *unit_name*)

obj_expr is an object expression

unit_name is a string representing the name of a distance unit (e.g. "km")

Return Value

Float

Description

The **SphericalObjectLen()** function always returns a value as calculated in a Latitude/Longitude non-projected coordinate system using great circle based algorithms.. A value of -1 will be returned for data that is in a NonEarth coordinate system since this data can't be converted into a Latitude/longitude coordinate system.

SphericalPerimeter () function

New function. Returns the perimeter of a graphical object.

Syntax

SphericalPerimeter(*obj_expr* , *unit_name*)

obj_expr is an object expression

unit_name is a string representing the name of a distance unit (e.g. "km")

Return Value

Float

Description

The **SphericalPerimeter()** function always returns a value as calculated in a Latitude/Longitude non-projected coordinate system using great circle based algorithms. A value of -1 will be returned for data that is in a NonEarth coordinate system since this data can't be converted into a Latitude/longitude coordinate system.

LegendFrameInfo() function

New function that returns information about a frame within a legend.

Syntax

LegendFrameInfo(window_id, frame_id, attribute)

window_id is a number that specifies which legend window you want to query.

frame_id is a number that specifies which frame within the legend window you want to query. Frames are numbered 1 to n where n is the number of frames in the legend.

attribute is an integer code indicating which type of information to return.

Return Value

Depends on the attribute parameter.

Attribute codes

FRAME_INFO_TYPE	Returns one of the following predefined constant indicating frame type: FRAME_TYPE_STYLE FRAME_TYPE_THEME
FRAME_INFO_MAP_LAYER_ID	Returns the id of the layer to which the frame corresponds.
FRAME_INFO_REFRESHABLE	Returns true if the frame was created without the Norefresh keyword. Always returns true for theme frames.
FRAME_INFO_POS_X	Returns the distance of the frames upper left corner from the left edge of the legend canvas (in paper units).
FRAME_INFO_POS_Y	Returns the distance of the frame's upper left corner from the topt edge of the legend canvas (in paper units).
FRAME_INFO_WIDTH	Returns the width of the frame (in paper units).
FRAME_INFO_HEIGHT	Returns the height of the frame (in paper units).
FRAME_INFO_TITLE	Returns the title of a style frame or theme frame .
FRAME_INFO_TITLE_FONT	Returns the font of a style frame title. Returns the default title font if the frame has no title or if it is a theme frame.
FRAME_INFO_SUBTITLE	Returns the subtitle of a style frame or theme frame.
FRAME_INFO_SUBTITLE_FONT	Same as title font.
FRAME_INFO_BORDER_PEN	Returns the pen used to draw the border.
FRAME_INFO_NUM_STYLES	Returns the number of styles in a frame. Zero if theme frame.
FRAME_INFO_VISIBLE	Returns true if the frame is visible (theme frames can be invisible).
FRAME_INFO_COLUMN	Returns the legend attribute column name as a string if there is one. Returns an empty string for a theme frame.
FRAME_INFO_LABEL	Returns the label expression as a string if there is one. Returns an empty string for a theme frame.

MapperInfo() function

A new attribute setting has been added to the MapperInfo() function.

The new MAPPER_INFO_DISPLAY_DMS attribute setting causes the MapperInfo() function to return a logical value indicating whether the map displays coordinates in degrees, minutes, seconds.

MapperInfo

Purpose

Returns coordinate or distance information about a Map window.

Syntax

MapperInfo(window_id , attribute)

window_id is an Integer window identifier

attribute is an Integer code, indicating which type of information should be returned

Return Value

Float, Logical, or String, depending on the attribute parameter

Description

The **MapperInfo()** function returns information about a Map window.

The *window_id* parameter specifies which Map window to query. To obtain a window identifier, call the **FrontWindow()** function immediately after opening a window, or call the **WindowID()** function at any time after the window's creation.

There are several numeric attributes that **MapperInfo()** can return about any given Map window. The attribute parameter tells the **MapperInfo()** function which Map window statistic to return. The attribute parameter should be one of the codes from the following table; codes are defined in MAPBASIC.DEF.

<u>attribute setting</u>	<u>MapperInfo() Return Value</u>
<code>MAPPER_INFO_DIST_CALC_TYPE</code> type of calculation perimeter, and area mapper. Corresponds to Set Map Distance Type.	Small integer, indicating to use for distance, length, calculations for Return value will be one of these:

`MAPPER_INFO_DIST_SPHERICAL`

`MAPPER_INFO_DIST_CARTESIAN`

`MAPPER_INFO_COORDSYS_CLAUSE_WITH_BOUNDS`

String result, indicating the window's CoordSys clause including the bounds.

`MAPPER_INFO_MOVE_DUPLICATE_NODES`

Small integer, indicating whether duplicate nodes should be moved when reshaping objects in this Map window.

If the value is 0, duplicate nodes are not moved.

If the value is 1, any duplicate nodes within the same layer will be move.

The attribute `MAPPER_INFO_MOVE_DUPLICATE_NODES` is 19.

LayerInfo

The `LayerInfo()` MapBasic function has been updated to return a new type for grid layers.

Syntax

```
LayerInfo( map_window_id , layer_number , attribute )  
map_window_id is a Map window identifier  
layer_number is the number of a layer in the current Map window  
(e.g. 1 for the top layer); to determine the number of layers in a Map  
window, call MapperInfo( )
```


attribute is a code indicating the type of information to return; see table below

LAYER_INFO_TYPE SmallInt value, indicating this layer's file type:
LAYER_INFO_TYPE_NORMAL for a normal layer;
LAYER_INFO_TYPE_COSMETIC for the Cosmetic layer;
LAYER_INFO_TYPE_IMAGE for a raster image layer;
LAYER_INFO_TYPE_THEMATIC for a thematic layer;
LAYER_INFO_TYPE_GRID for a grid image layer.

WindowInfo

Snap mode settings for a particular window can be queried by using new attribute parameters in the MapBasic WindowInfo() function.

Syntax:

```
WindowInfo ( window_spec , attribute )
```

New Attributes:

WIN_INFO_SNAPMODE

Returns a logical value. TRUE if snap mode is on. FALSE if snap mode is off.

The value for WIN_INFO_SNAPMODE is 19.

WIN_INFO_SNAPTHRESHOLD

Returns a SmallInt value representing the pixel tolerance.

The value for WIN_INFO_SNAPTHRESHOLD is 20.

Set Map

Purpose

Modifies an existing Map window.

Syntax

The main part of a **Set Map** statement has the following syntax:

Set Map

```
[ Window window_id ]  
[ Center ( longitude, latitude ) [ Smart Redraw ] ]  
[ Clipping { Object clipper | Off | On } ]  
[ Zoom { zoom_distance [ Units dist_unit ] | Entire [ Layer  
layer_id ] } ]  
[ Preserve { Scale | Zoom } ]  
[ Display { Scale | Position | Zoom } ]  
[ Order layer_id, layer_id [ , layer_id ... ] ]  
[ Pan pan_distance [ Units dist_unit ]  
{ North | South | East | West } [ Smart Redraw ] ]  
[ CoordSys... ]  
[ Area Units area_unit ]  
[ Distance Units dist_unit ]  
[ Distance Type { Spherical | Cartesian } ]  
[ XY Units xy_unit ]  
[ Scale screen_dist [ Units dist_unit ] For map_dist [ Units  
dist_unit ] ]  
[ Redraw { On | Off } ]
```

```

[ Move Nodes { value | Default } ]
[Inflect num_inflections at color:expr [, color:expr, ... ] ]
  [ Contrast contrast_value ]
  [ Brightness brightness_value ]
  [ GrayScale { On | Off } ]
LAYERCLAUSE
LAYERCLAUSE . . .

```

window_id is the Integer window identifier of a Map window.

If a Distance Type is entered, it can either be **Spherical** or **Cartesian**. All distance, length, perimeter, and area calculations for objects contained in the Map Window will be performed using one of these calculation methods. Note that if the **Coordsys** of the Map Window is NonEarth, then the calculations will be performed using **Cartesian** methods regardless of the option chosen, and if the **Coordsys** of the Map Window is Latitude/Longitude, then calculations will be performed using **Spherical** methods regardless of the option chosen.

The "Move Node value" can be 0 or 1. If the value is 0, duplicate nodes are not moved. If the value is 1, any duplicate nodes within the same layer will be moved. If a "Move Node value" is specified, that window is considered to be using a custom value. To return to using the default (from the mapper preference), "Move Nodes Default" can be specified.

Set Window

```

Set Window window_id
  [ Position ( x , y ) [ Units paper_units ] ]
  [ Width win_width [ Units paper_units ] ]
  [ Height win_height [ Units paper_units ] ]
  [ Font ... ]
  [ Min | Max | Restore ]
  [ Front ]
  [ Title { new_title | Default } ]
  [ Help [ { File help_file | File Default | Off } [
Permanent ] ]
          [ Contents ] [ ID context_ID ] [ { Show | Hide
} ]
  [ ScrollBars { On | Off } ]
  [ Smart Pan { On | Off } ]
  [ Autoscroll { On | Off } ]
  [ Parent HWND ]
  [ ReadOnly | Default Access ]
  [ Table table_name Rec record_number ]
  [ Show | Hide ]
  [ SysMenuClose { On | Off } ]
  [ Snap [ Mode { On | Off } ] [ Threshold { pixel_tolerance |
Default } ]
  [ Printer { Default |
          [ Name printer_name ]
          [ Orientation { Portrait | Landscape } ]
          [ Copies number ]
} ]

```

Where:

printer_name identifies a printer. The printer can be local or networked to the computer on which MapInfo Professional is running. *number* is the number of copies of a print job that should be sent to the printer.

WindowInfo()

...

New Attribute codes:

WIN_INFO_PRINTER_NAME

Returns string value with printer identifier (e.g. \\DISCOVERY\HP4_DEVEL)

WIN_INFO_PRINTER_ORIENT

Returns WIN_PRINTER_PORTRAIT or WIN_PRINTER_LANDSCAPE

WIN_INFO_PRINTER_COPIES

Returns integer number of copies.

Relief Shade statement

A new statement. Adds relief shade information to an open grid table.

Syntax

Relief Shade

```
Grid tablename  
Horizontal xy_plane_angle  
Vertical incident_angle  
Scale z_scale_factor
```

tablename is the alias name of the grid to which relief shade information is being calculated.

xy_plane_angle is the direction angle, in degrees, of the light source in the horizontal or *xy* plane. An *xy_plane_angle* of zero represents a light source shining from due East. A positive angle places the light source counterclockwise, so to place the light source in the NorthWest, set the *xy_plane_angle* to 135.

incident_angle is the angle of the light source above the horizon or *xy* plane. An *incident_angle* of zero represents a light source right at the horizon. An *incident_angle* of 90 places the light source directly overhead.

z_scale_factor is the scale factor applied to the z-component of each grid cell. Increasing the *z_scale_factor* enhances the shading effect by exaggerating the vertical component. This can be used to bring out more detail in relatively flat grids.

Set Date Window statement

A new statement. Display a date window that converts two-digit input into four digit years. It also allows you to change the default to one that best suits your data.

Syntax

```
Set Date Window { nYear | Off }
```

nYear A SmallInt from 0 to 99 that specifies the year equal to and above which is the current century (20th) and below which is the next century (21st).

Off Turns date windowing off. Two digit years will be converted to the current century (based on system time/calendar settings).

Description

From the MapBasic window, the session setting will be initialized from the Preference setting and updated when the preference is changed. Running the Set Date Window command from the MapBasic window will change the behavior of input, but will not update the System Preference that is saved when MIPro exits.

The session setting is affected by running Set Date Window in the MapBasic window, in any workspace file including Startup.WOR, and any

Integrated Mapping app that runs the command via the MapInfo Application interface.

When the Set Date Window command is run from within a MapBasic program (also as Run Command) only the program's local context will be updated with the new setting. The session and preference settings will remain unchanged. The program's local context will be initialised from the session setting. This is similar to how number and date formatting works. They are set/accessed per program if a program is running, otherwise they set/access global settings.

Note: All assignments to date variables in previous MapBasic versions will be affected. They can take two-digit year expressions and will have to use the date window to determine which century should be used.

```
Year(date_expr)
StringToDate(datestring)
FormatDate$(date_expr)
```

Set Map statement

A new Relief clause has been added. The optional Relief clause turns relief shading on/off for a grid. The grid must have relief shade information calculated for it for it for this clause to have any effect. Relief shade information can be calculated for a grid with the Relief Shade command.

Inflection colors settings are persisted here. These override the inflection color:value pairs that are stored in the grid file.

Set Map

```
Layer { layer_id | layer_name }
  { Inflect num_inflections [ by percent ]
    at color:expr [, color:expr, ... ] }
  [ Round rounding_factor ]
  [ Contrast contrast_value ]
  [ Brightness brightness_value ]
  [ GrayScale { On | Off } ]
  [ Relief { On | Off } ]
```

A new Set Map XY Units clause has been added to the Set Map statement.

If the XY Units are in degrees, the Display Decimal clause specifies whether to display in decimal degrees (**On**) or in degrees, minutes, seconds (**Off**).

```
Set Map
  [ XY Units "degree" ]
  [ Display Decimal { On | Off } ]
```

A new distance clause has been added.

Set Map

```

    [ Window window_id ]
    [ Center ( longitude, latitude ) [ Smart Redraw ] ]
    [ Clipping { Object clipper | Off | On } ]
    [ Zoom { zoom_distance [ Units dist_unit ] | Entire [ Layer
layer_id ] } ]
    [ Preserve { Scale | Zoom } ]
    [ Display { Scale | Position | Zoom } ]
    [ Order layer_id, layer_id [ , layer_id ... ] ]
    [ Pan pan_distance [ Units dist_unit ]
      { North | South | East | West } [ Smart Redraw ] ]
    [ CoordSys... ]
    [ Area Units area_unit ]
    [ Distance Units dist_unit ]
    [ Distance Type dist_type ]
    [ XY Units xy_unit ]
    [ Scale screen_dist [ Units dist_unit ] For map_dist [
Units dist_unit ] ]
    [ Redraw { On | Off } ]
    [ Inflect num_inflections at color:expr [ , color:expr, ... ] ]
      [ Contrast contrast_value ]
      [ Brightness brightness_value ]
      [ GrayScale { On | Off } ]
    LAYERCLAUSE
    LAYERCLAUSE . . .

```

dist_type is a short integer value representing the type of algorithm to use for all distance, length, perimeter, and area calculations of the map. Value will be one of the following:

```

    MAPPER_INFO_DIST_DEFAULT
    MAPPER_INFO_DIST_GREAT_CIRCLE
    MAPPER_INFO_DIST_CARTESIAN

```

Setting Printer Control

Allow MapBasic to control of the printer used to print from MapInfo.

By default, MapInfo prints using the global printer device initialized to the default Windows printer. The new **Name clause** allows an application, workspace, or the MapBasic window to override this printer selection for individual document windows and selects another available printer on the system. Several settings for the printer can also be controlled by using additional command clauses. These overrides are saved with the workspace commands for the affected windows, so they will be reapplied when the workspace is reopened.

An override can be removed from a window by running a **Set Window** with **Printer Default**.

Extensions to the **Set Window** command and the **WindowInfo()** function provide the necessary control.

Addition to the Set Window statement allow resetting

```

Set Window window_id ...
  [ Printer { Default |
              [ Name printer_name ]

```

```
[ Orientation { Portrait | Landscape } ]  
[ Copies number ]  
} ]
```

Snap Mode and the Set Window statement

Each window now has its own snap mode stat and snap mode and tolerance is now settable for each mapper and layout window. Snap mode and tolerance is saved in the workspace so the snap settings are saved for each window.

From the MapBasic window you can snap to a particular pixel tolerance for a given window, set snap back to the default snap tolerance for a given window, Retrieve the current snap tolerance for a given window, turn snap on/off for a given window and retrieve information about whether snap is on/off for a window

Snap mode settings are set using the Set Window MapBasic statement. Snap mode settings for a particular window can be queried by using new attribute parameters in the WindowInfo() function.

Syntax:

```
Set Window window_id
    [ Position ( x , y ) [ Units paper_units ] ]
    [ Width win_width [ Units paper_units ] ]
    [ Height win_height [ Units paper_units ] ]
    [ Font ... ]
    [ Min | Max | Restore ]
    [ Front ]
    [ Title { new_title | Default } ]
    [ Help [ { File help_file | File Default | Off } [
Permanent ] ]
    [ Contents ] [ ID context_ID ] [ { Show | Hide
} ]
    [ ScrollBars { On | Off } ]
    [ Smart Pan { On | Off } ]
    [ Autoscroll { On | Off } ]
    [ Parent HWND ]
    [ ReadOnly | Default Access ]
    [ Table table_name Rec record_number ]
    [ Show | Hide ]
    [ SysMenuClose { On | Off } ]
    [ Snap [ Mode { On | Off } ] [ Threshold { pixel_tolerance |
Default } ]
```

From the MapBasic window, the snap mode tolerance can be set for a particular window.

```
Set window win snap [mode on/off]
                    [Threshold value|Default]
```

WindowInfo() function

Snap mode settings for a particular window can be queried by using new attribute parameters in the WindowInfo() function. Also, new printer control attributes have been added.

Syntax:

WindowInfo (window_spec , attribute)

New Attributes:

WIN_INFO_SNAPMODE
Returns a logical value. TRUE if snap mode is on. FALSE if snap mode is off.

The value for WIN_INFO_SNAPMODE is 19.

WIN_INFO_SNAPTHRESHOLD
Returns a SmallInt value representing the pixel tolerance.
The value for WIN_INFO_SNAPTHRESHOLD is 20.

WIN_INFO_PRINTER_NAME
Returns string value with printer identifier (e.g. \\DISCOVERY\HP4_DEVEL)

WIN_INFO_PRINTER_ORIENT
Returns WIN_PRINTER_PORTRAIT or WIN_PRINTER_LANDSCAPE

WIN_INFO_COPIES
Returns integer number of copies