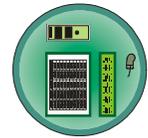


The CEDRA Corporation's COMMAND OF THE MONTH

A monthly information bulletin

November 2007

FEATURED COMMAND
CEDRA-DataEditor
Multi-Line Map Tips



Application Description

When the user wishes to quickly display an attribute value for a feature, the Map Tip functionality within ArcMap® can be invoked. By doing so, when the user moves the cursor over a feature, a map tip will appear containing the value for the primary display field.

For those who are unfamiliar with map tips, the ArcMap Desktop Help describes map tips as a means to quickly present information about map features to those viewing data with ArcMap. Map Tips are more than simply labeling features with text. Map tips pop up as the user pauses the mouse pointer over a feature. This provides a quick method to see the name of a feature or a particular piece of information about it, without having to use ArcMap's Identify tool, which gives you all of the feature's attributes when you click on a feature.

By specifying the primary display field, the user is able to control which attribute (field) will appear in the map tip. The primary display field is defined under the Fields tab within the [Properties...] dialog box, which is displayed when the user right-clicks on the name of the layer in the Table of Contents and the selects the [Properties...] menu item.

The limitation of this functionality is that the user can only display a single attribute value within the map tip. Additionally, there is no prefix to identify the attribute being displayed.

Given this limitation, it would be nice if there was a way to be able to display a



Figure 1
CEDRA-DataEditor-Tools Toolbar

multi-line map tip. Even better would be the ability to include a prefix to better identify the values being displayed.

The CEDRA Solution

To address the issue of displaying multi-line map tips, the user can activate the CEDRA-DataEditor-Tools toolbar, see Figure 1. The tools contained in this toolbar are available only for ArcGIS® users. Specifically, the right-most tool shown in Figure 1 will be the command that we will be discussing in this issue of Command of the Month.

Command Of The Month bulletin

This month's issue discusses the display of multi-line map tips, with prefixes, when the user pauses the cursor over a specific feature.

Map Tips - Overview

The [Map Tips] tool  provides the user the ability to move the cursor over a desired feature and have selected attribute or field values appear in the form of a multi-line map tip.

The attribute values that appear in the map tip are those which appear in the

Theme Attribute Data File, *themes.txt*, which the CEDRA-DataEditor software processes. The February 2005 issue of Command of the Month describes the structure of the Theme Attribute Data File and the operation of the CEDRA-DataEditor software.

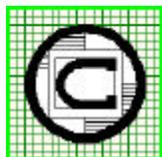
Essentially, the Theme Attribute Data File is an ASCII based file that the user creates using any text editor or word processor. The name of this file is *themes.txt* and typically resides in the same directory or folder that the ArcMap document file (*.mxd) resides in.

Shown in Figure 2 is a sample Theme Attribute Data File. Within this sample file, one layer is defined and within this layer there are nine attributes which the user can edit with the [Edit Feature Attributes] tool  or display in a map tip with the [Map Tips] tool.

Note that there is no limit to the number of layers or attributes that can be defined within the Theme Attribute Data File. For simplicity, only one layer has been included in the sample shown in Figure 2.

Map Tips - Operation

1. Select the [Map Tips] tool.
2. Move the cursor on top of a feature which appears in a layer that appears in the Theme Attribute Data File and maintain the cursor position. The tool will now begin a feature search.



™ The CEDRA Corporation

Total CADD for Engineers™
Bridging Engineering and GIS™

Phone: 585-232-6998
Fax: 585-262-2042

151 Sully's Trail - Suite 6
Pittsford, New York 14534

E-mail: cedra@cedra.com
URL: <http://www.cedra.com>

Copyright © 2007 The CEDRA Corporation. The CEDRA logos, CEDRA-AVseries, CEDRA-AVcad, CEDRA-AVcogo, CEDRA-AVparcel, CEDRA-AVland, CEDRA-AVsand, CEDRA-AVwater, CEDRA-DataEditor and CEDRA-DxfExport are trademarks of The CEDRA Corporation. ArcView® GIS, and ArcGIS are registered trademarks of Environmental Systems Research Institute, Inc. All World Rights Reserved.

```

/*
/* Command denoting Start of
/* a Layer Definition
/*
THEME
/*
/* Name of the Layer as it
/* appears in ArcMap's Table
/* of Contents
/*
Parcels
/*
/* List of Attributes in the
/* Layer to be Edited and
/* Displayed as a Map Tip
/*
/* Att. 1: Lot Number
/*
LOTNUM
Lot Number:
/*
/* Att. 2: Block Number
/*
BLOCKNUM
Block Number:
REMEMBER_LAST
/*
/* Att. 3: Plat Name
/*
PROJECT
Plat Name:
REMEMBER_LAST
/*
/* Att. 4: House Number
/*
HOUSE
House Number:
/*
/* Att. 5: Street Direction
/*
DIR
Street Direction:
/*
/* Command denoting display
/* a choice list combo-box
/* rather than a data line
/*
CHOICES

N
S
E
W
NE
SE
SW
NW
END
REMEMBER_LAST

/*
/* Att. 6: Street Name
/*
NAME
Street Name:
REMEMBER_LAST
/*
/* Att. 7: Street Type
/*
TYPE
Street Type:
/*
/* Command denoting display a choice list combo-box
/* rather than a data line
/*
CHOICES

AVE
BAY
BLVD
CIRC
COVE
CT
DR
HILL
LANE
PARK
PATH
PKY
PL
ROAD
ST
TERR
TRL
WAY
END
REMEMBER_LAST
/*
/* Att. 8: Street Address
/*
ADDRESS
Parcel Address:
/*
/* Command denoting attribute value is to be
/* computed by evaluating the following equation
/*
ATTRIBUTE_EQUATION
TRIM(HOUSE)+" "+TRIM(DIR)+" "+TRIM(NAME)+" "+TRIM(TYPE)
/*
/* Att. 9: Section Number
/*
SECTION
Section Number (T/R/S):
REMEMBER_LAST
/*
/* Command denoting End of Layer Definition
/*
END

```

Figure 2 - Sample Theme Attribute Data File (themes.txt)
containing 1 Layer with 9 attributes that can be edited or displayed in a Map Tip

Once the program finds a feature which satisfies the above conditions, a map tip will appear above and to the right of the cursor position. Shown in Figure 3 is a sample multi-line map tip.

```
Lot Number: 7
Block Number: 3
Plat Name: WALNUT RIDGE ESTATES
House Number: 2308
Street Direction:
Street Name: SUMMIT
Street Type: PL
Parcel Address: does not exist
Section Number (T/R/S): 140207
```

Figure 3
Sample Multi-Line Map Tip

Note that the attribute values which are displayed are prefixed with the attribute label that appears in the Theme Attribute Data File. This is the data line that appears below the name of the attribute in the Theme Attribute Data File. For example, the attribute LOTNUM is assigned the Attribute Label, LotNumber:. As such, this label prefixes the attribute value shown in the map tip above.

Notes

- a. The Theme Attribute Data File is read when ArcMap is invoked. If the user modifies the Theme Attribute Data File and wishes to have the file reread, the [Reload Theme Attribute Data File] tool  can be used. This tool will read the *themes.txt* file that resides in the current working directory.
- b. To set the current working directory, the [Set Working Directory] tool  can be used.
- c. Data lines which begin with the /* characters, see Figure 2, denote a comment line. Comment lines are not processed and appear in the Theme Attribute Data File to improve the readability of the file and provide information to others who may wish to view the file.

- d. The abbreviation *Att.*, which appears in some of the comment lines, see Figure 2, represents the word, Attribute. As used in this publication, the words Attribute and Field are synonymous.
- e. When specifying the name of the attribute to be processed, the user can enter either: (a) the true name of the attribute (field) or (b) the name of the alias that has been assigned to the attribute, if any.
- f. The order in which the attributes are listed in the Theme Attribute Data File for a layer will control the order in which they are displayed in the [Edit Feature Attributes] tool's dialog box and within the map tip.

The following notes do not pertain to the [Map Tips] tool but do pertain to the [Edit Feature Attributes] tool. They are offered only to help clarify the information shown in Figure 2. The reader is referred to the text which follows the Summary section for a full discussion on the structure of the Theme Attribute Data File.

- g. The REMEMBER_LAST command, which appears below the Attribute Label in Figure 2, denotes that the last value entered by the user for the attribute should be used as the default value the next time a feature is selected and whose attribute value is NULL or blank.
- h. The CHOICES command, as shown in Figure 2, indicates that the user wishes to have a choice list combo-box displayed, rather than a data line field. Below the command, the user enters the available options which are to appear in the choice list combo-box. The user is only able to select the options presented in the choice list combo-box, any other entry would result in an error message being displayed. If a blank line is a valid option, then the user should enter a data line with at least one blank character (space). The END com-

mand must be entered to denote the end of the list of options.

- i. The ATTRIBUTE_EQUATION command, as shown in Figure 2, indicates that the value for the attribute is to be computed and not entered by the user. The equation which appears below the command is evaluated and the result is used as the value to be assigned to the attribute. The equation is evaluated when the user: (a) selects the OK button, and (b) when the user depresses the Enter key in a data line field in the [Edit Feature Attributes] tool's dialog box. When the [Map Tips] tool is used, the equation is not evaluated, the command simply displays the current value assigned to the attribute.
- j. The TRIM function, which appears in the attribute equation, indicates that the value which appears within the () characters is to have any leading and/or trailing blanks removed from the value and if used in conjunction with the + operator, will remove the last or first character from the attribute value it is being added to if the character is a blank (space). This is useful in removing double blank characters which may appear in the case when address components are being added together and one of the components is blank (has no data).

Summary

The CEDRA-DataEditor software is included with all of the CEDRA-AVseries software, as well as, licensed as a stand-alone software package. It is extremely useful for those involved in entering and editing data.

It is highly recommended that if you have not used the CEDRA-DataEditor software, you consider doing so. The [Edit Feature Attributes] tool  and the [Map Tips] tool  are very easy to use and provide a highly efficient means of maintaining and viewing data.

Theme Attribute Data File Description

Presented below is a short description of the Theme Attribute Data File. The name of this file is always *themes.txt* and it should reside in the directory where the ArcMap document file resides. This location is referred to as the current working directory.

The *themes.txt* file is a sequential ASCII based file that contains a list of themes and their attributes, which the user can edit with the [Edit Feature Attributes] tool



. The CEDRA-DataEditor software supports two types of database design for a theme.

- The *first type* is the one in which all of the attributes are stored in the theme (layer).
- The *second type* is one in which some attributes are stored in the theme, and some are stored in associated tables. Under this design type, there are *two* attributes in the theme, which link a record (feature) in the theme to the appropriate record in the associated table. The *first* attribute represents a unique identifier (ID), that links a record in the theme with a record in the associated table, while the *second* contains an attribute whose value points to the associated table.

It is noted that it is possible for the Theme Attribute Data File to contain a mixture of these database designs. That is, the Theme Attribute Data File does not have to contain themes of a single database design type, but rather, it can contain both types, if need be.

A. Database Design Type 1

All Attributes Stored in a Layer

Shown in Table 1 is a sample file that contains a single theme with 13 attributes, which are available for editing. In this table, the user will notice the following:

<i>Comment lines</i>	{ /* /* CEDRA Point Theme /* /*
<i>Number of labels per dialog box page</i>	NLINES 15 /*
<i>Theme command</i>	THEME
<i>Theme name</i>	L_0p.n /*
<i>Attribute name w/o units</i>	MAP
<i>Attribute label</i>	Map Number: /*
<i>Attribute name w/o units</i>	PNT
<i>Attribute label</i>	Point Number: /*
<i>Attribute name w/ units</i>	X 1
<i>Attribute label</i>	Point X - ft (m): /*
	Y 1
	Point Y - ft (m): /*
	Z 1
	Point Elevation - ft (m): /*
	PTCODE
	Point Code (max 8 chrs.): /*
	PTDESC
	Point Description (max 40 chrs.): /*
	MRK
	Point Marker: /*
	SCL
	Point Scale Factor: /*
	MOD
	Point Mode: /*
	CLR
	Point Color: /*
	PEN
	Pen Weight: /*
<i>End of Theme command</i>	END

1. The characters /* can be used to introduce comment lines, and must appear in columns 1 and 2 of a data line. Comment lines can appear as often as desired and anywhere

within the file. Comment lines are used to make the file easier to read. There is no limit to the number of characters in a comment line.

2. The NLINES command designates how many attributes are to be displayed per page of the editing dialog box. The command should be followed with a number, between 1 and n, designating said number of attributes. The maximum value of n is 20 for ArcView® GIS users and 30 for ArcGIS users.
3. The THEME command designates the beginning of a new theme.
4. The THEME command should be followed by the name of the theme as it appears in the View's Table of Contents.

Below the name of the theme should be the attributes that the user can edit with the [Edit Feature Attributes] tool. Two data lines, at a minimum, are used to define an attribute.

5. The first attribute data line contains the name of the attribute, as it appears in the theme's attribute table, and optionally, a code identifying the attribute's units of measure. These items must be separated by at least one blank space.
 - In specifying the name of the attribute, the user can specify either the source name of the attribute or its alias, if any has been assigned. In addition, the name of the attribute should not contain any blank characters.
 - The default units setting for an attribute is no units, that is, the value for an attribute is unit-less. The valid attribute units of measure codes are 0 through 5, and correspond to:
 - 0 No units apply to the attribute
 - 1 Feet or Meters
 - 2 Inches or Millimeters
 - 3 Acres or Hectares
 - 4 Miles or Kilometers
 - 5 Square Feet or Square Meters

6. Note that at least one space or blank character must separate the name of the attribute from the units value. If no units value is specified, the units setting will default to be unit-less.

The second data line contains the attribute label that should appear in the multi-input dialog box to describe the attribute to the user. For example, in the sample presented in Table 1, the label Point Scale Factor: is used to describe the SCL attribute.

7. The END command is used to terminate the attribute specification for the theme. The order in which the attributes are listed between the THEME and END commands will control how the attributes appear in the multi-input dialog box.

The general file structure for a theme under this database design is as follows:

```
THEME
theme_name
attribute_1 optional_units_setting
attribute_1_label
attribute_2 optional_units_setting
attribute_2_label
.
.
.
attribute_n optional_units_setting
attribute_n_label
END
```

Repeat the above block of data (from THEME to END) for every theme to be available to the [Edit Feature Attributes] tool.

B. Database Design Type 2

Attributes Stored in a Layer and an Associated Table

Shown in Table 2 is a sample file that contains a single theme with 6 attributes available for editing with two associated tables that contain 5 and 11 attributes respectively. When this database design is employed, the structure of the

Theme Attribute Data File is essentially the same as under the first database design with the following exceptions:

1. The first attribute defined must be the attribute containing the unique identifier that links a record in the theme (a feature) with a record in the associated table. As such, this attribute must exist in both the theme and the associated table(s) and must be of the same name and type, either character or numeric.

In the sample shown in Table 2, the attribute FACILITY_ID is the first attribute defined, and as such, is used as the unique identifier (ID) to link a record in the theme with a record in the associated table. That is to say, there is a one to one correspondence with a record in the theme (a feature) with a record in the associated table. Specifically, both records will have the same value for the FACILITY_ID attribute.

2. The last attribute in the theme, that is being defined, prior to defining the attributes in the associated table, must be the attribute whose value denotes the name of the appropriate associated table.

The PW_TYPE attribute is the last theme attribute defined, and as such, is used to point to the appropriate associated table. In the sample of Table 2, the PW_TYPE attribute has two possible values DINLET and DMANHOLE, which reference the DINLAT_DAT and DMANHOLE_DAT tables respectively. No other values for the PW_TYPE attribute are valid.

3. The order of theme attributes between the first and last attributes should be in the order that the user wishes them to appear in the multi-input dialog box.
4. Following the specification of the last attribute in the theme, the user defines the associated tables.

Table 2 - Database Design Type 2

	<pre> /* Theme with Two Associated Tables called: /* DINLET_DAT and DMANHOLE_DAT /* The associated tables are referenced by the /* PW_TYPE attribute of the Theme that has two /* allowable values DINLET and DMANHOLE. /* The attribute that links the Theme with the /* associated table is FACILITY_ID /* NLINES 20 /* THEME ST_STORMpn.shp /* FACILITY_ID FACILITY_ID: /* ANGLE ANGLE: /* SOURCE SOURCE: /* PNT PNT: /* QUALITY QUALITY (1, 2, 3): /* PW_TYPE PW_TYPE: /* /* Associated Table 1 with 5 attributes /* /* The line below contains the table command /* and the attribute value DINLET for the /* attribute PW_TYPE TABLE DINLET /* The line below contains the name of the /* associated table with which DINLET is /* associated DINLET_DAT FACILITY_ID FACILITY_ID: </pre>
<i>Number of labels per dialog box page</i>	NLINES 20
<i>Theme command</i>	THEME ST_STORMpn.shp
<i>First Theme attribute</i>	FACILITY_ID
<i>First attribute label</i>	FACILITY_ID:
	ANGLE ANGLE:
	SOURCE SOURCE:
	PNT PNT:
	QUALITY QUALITY (1, 2, 3):
<i>Last attribute</i>	PW_TYPE
<i>Last attribute label</i>	PW_TYPE:
	TABLE DINLET
<i>Table command and last Theme attribute value</i>	TABLE DINLET
	DINLET_DAT
<i>Name of the associated table</i>	DINLET_DAT
<i>Same as the First Theme attribute</i>	FACILITY_ID
<i>Same as the First attribute label</i>	FACILITY_ID:

- Following the specification of the last attribute, the TABLE command is used to define the associated table. The TABLE command requires one argument, that being, the value of the last attribute that was defined that references the associated table.
- Following the TABLE command is the name of the associated table as it appears under the Source tab in the Table of Contents.
- Below the name of the associated table, the user identifies the attributes that appear in the associ-

Table 2 - Database Design Type 2 (cont.)

<p><i>Table command and last Theme attribute value</i></p> <p><i>Name of the associated table Same as the First Theme attribute Same as the First attribute label</i></p>	<pre> SUB_TYPE SUB_TYPE: INLET_HEIGHT 2 INLET_HEIGHT - in (mm): INLET_LENGTH 2 INLET_LENGTH - in (mm): INLET_WIDTH 2 INLET_WIDTH - in (mm): /* /* Associated Table 2 with 11 attributes /* /* The line below contains the table command /* and the attribute value DMANHOLE for /* the Theme attribute PW_TYPE TABLE DMANHOLE /* The line below contains the name of the /* associated table with which DINLET is /* associated DMANHOLE_DAT FACILITY_ID FACILITY_ID: SUB_TYPE SUB_TYPE: DEPTH 1 DEPTH - ft (m): ELEV_INVERT 1 ELEV_INVERT - ft (m): ELEV_RIM 1 ELEV_RIM - ft (m): BARREL_DIAM 2 BARREL_DIAM - in (mm): MH_LENGTH 2 MH_LENGTH - in (mm): MH_WIDTH 2 MH_WIDTH - in (mm): MH_MATERIAL MH_MATERIAL LID_MATERIAL LID_MATERIAL LID_DIAMETER 2 LID_DIAMETER - in (mm): END </pre>
--	--

ated table, which the user can edit with the [Edit Feature Attributes] tool. The attributes in an associated table are defined in the same manner as those defined for a theme.

8. Following the specification of the last attribute in an associated table, the user can define additional associated tables by entering additional TABLE commands and pertinent data, or terminate the specification

for the theme by entering the END command.

The general file structure for a theme under this database design is as follows:

```
THEME
theme_name
attribute_1 optional_units_setting
attribute_1_label
attribute_2 optional_units_setting
attribute_2_label
.
.
attribute_n optional_units_setting
attribute_n_label
TABLE table_attribute
table_name
table_attribute_1 optional_units_setting
table_attribute_1_label
table_attribute_2 optional_units_setting
table_attribute_2_label
.
.
table_attribute_n optional_units_setting
table_attribute_n_label
TABLE table_attribute
table_name
table_attribute_1 optional_units_setting
table_attribute_1_label
table_attribute_2 optional_units_setting
table_attribute_2_label
.
.
table_attribute_n optional_units_setting
table_attribute_n_label
.
.
END
```

Note that the definition of a table is terminated by the presence of another TABLE command, or the detection of the END command.

Repeat the above block of data (from THEME to END) for every theme to be available to the [Edit Feature Attributes] tool.

There is no limit to the number of associated tables or number of attributes in an associated table that can be defined. In addition, there is no limit to the number of themes or attributes that can appear in this file.

9. The NLINES command designates how many attributes are to be displayed per page of the editing dia-

log box. The command should be followed with a number, between 1 and n, designating said number of attributes. The maximum value of n is 20 for ArcView® GIS users and 30 for ArcGIS users.

C. Additional Commands

Enhanced Attribute Controls

In addition to the commands described above, the CEDRA-DataEditor software provides the user the option of employing the following commands:

```
RANGE
CHOICES
DEFAULT
REMEMBER_LAST
ATTRIBUTE_EQUATION
```

These commands provide the user additional control over the values that are assigned to an attribute, and as such increase the Quality Control aspect of data maintenance. Depending upon the specific command, the location of where the command appears within the Theme Attribute Data File will vary. Presented below is a discussion of the commands, listed above, along with samples denoting the location of where the command should appear in the Theme Attribute Data File.

Command Option Number 1:

Below the attribute label, the user has the option of entering the **RANGE** or **CHOICES** commands.

The **RANGE** command applies to attributes, which are of *number* type, and can be used to indicate the valid range of numbers, inclusive, that an attribute can be assigned.

For example, the MAP attribute defined below can be assigned any number between 1 and 99999, inclusive. Any value that is entered for this attribute, and which is outside this range will cause an error message to be displayed. The range of numbers is defined on the same

data line as the command with at least one space separating the values and the command.

```
/*
MAP
Map Number :
RANGE 1 99999
```

The **CHOICES** command can be used for any attribute, when the user may have a set of allowable values that can be assigned to an attribute. For example, the SOURCE attribute, shown in the sample below, can be assigned either ATLAS or AS-BUILT. No other value can be assigned to this attribute.

Below the **CHOICES** command the user lists the possible values that can be assigned to an attribute, using a distinct data line for each value. Following the last value, the END command *must* be entered to terminate the **CHOICES** command. Note that this END command should not be confused with the END command used to terminate a theme definition.

```
/*
SOURCE
SOURCE :
CHOICES
ATLAS
AS-BUILT
END
```

There is no limit to the number of values that can be listed under the **CHOICES** command. It is possible for the user to enter a blank line under the **CHOICES** command to indicate that the attribute can be assigned a blank or empty string.

When this command is used a choice list combo-box will appear in the [Edit Feature Attributes] tool's dialog box containing the options in the same order as they appear in the Theme Attribute Data File.

The user simply uses the drop-down list to select the appropriate value for the attribute. In so doing the potential for any data entry error is eliminated. The **CHOICES** command can be used for

both *numeric* and *character* type attributes.

Note that an attribute can be assigned either the **RANGE** or the **CHOICES** command, but not both.

Command Option Number 2:

Below the **RANGE** or **CHOICES** command, the user has the option of entering the **DEFAULT** command. The **DEFAULT** command indicates the default or initial value of an attribute that should be assigned to:

- a feature that is copied into a theme identified in the Theme Attribute Data File or
- a feature whose attribute value has not been defined.

```
/*
SOURCE
SOURCE:
CHOICES
ATLAS
AS-BUILT
END
DEFAULT ATLAS
/*
MAP
Map Number :
DEFAULT 1
```

In the example above, the default value that would be assigned to the **SOURCE** attribute is **ATLAS**, and for the **MAP** attribute, the default value would be one.

The default value is used when the value for an attribute of string or character type is undefined or blank, and for an attribute of number type, when the value is **NULL** or zero. The **DEFAULT** command must appear after the **RANGE** or **CHOICES** commands, not before.

In addition, the **DEFAULT** command will recognize various key-codes, such as:

```
RETURN_LENGTH,
RETURN_AREA and
RETURN_AREA_UNITS.
```

These specific key-codes will result in the **DEFAULT** command using as the default value for an attribute the:

- Length or perimeter of a feature (**RETURN_LENGTH**),
- Area of a feature in square feet or square meters (**RETURN_AREA**), or
- Area of a feature in acres or hectares (**RETURN_AREA_UNITS**).

To use these key-codes, the user simply enters the appropriate key-code after the **DEFAULT** command with at least one space separating the two items. As an example, the user could enter:

```
DEFAULT RETURN_AREA
```

to have the CEDRA-DataEditor software compute and display, as the default value for an attribute, the area of a feature in square feet or square meters depending upon the current units of measure property setting.

The various forms in which the **DEFAULT** command can appear in are as follows:

```
DEFAULT some_value
DEFAULT ADDONE
DEFAULT CURRENT_DATE
DEFAULT RETURN_LENGTH
DEFAULT RETURN_AREA
DEFAULT RETURN_AREA_UNITS
DEFAULT RETURN_X
DEFAULT RETURN_Y
DEFAULT RETURN_X_DD
DEFAULT RETURN_Y_DD
DEFAULT RETURN_X_DMS
DEFAULT RETURN_Y_DMS
DEFAULT RETURN_X_LONG
DEFAULT RETURN_Y_LAT
DEFAULT RETURN_XS
DEFAULT RETURN_YS
DEFAULT RETURN_XS_DD
DEFAULT RETURN_YS_DD
DEFAULT RETURN_XS_DMS
DEFAULT RETURN_YS_DMS
DEFAULT RETURN_XS_LONG
DEFAULT RETURN_YS_LAT
DEFAULT RETURN_XM
DEFAULT RETURN_YM
```

```
DEFAULT RETURN_XM_DD
DEFAULT RETURN_YM_DD
DEFAULT RETURN_XM_DMS
DEFAULT RETURN_YM_DMS
DEFAULT RETURN_XM_LONG
DEFAULT RETURN_YM_LAT
DEFAULT RETURN_XE
DEFAULT RETURN_YE
DEFAULT RETURN_XE_DD
DEFAULT RETURN_YE_DD
DEFAULT RETURN_XE_DMS
DEFAULT RETURN_YE_DMS
DEFAULT RETURN_XE_LONG
DEFAULT RETURN_YE_LAT
```

A description of the key-codes shown above is as follows:

The key-code **ADDONE** results in the **DEFAULT** command using as the default value for an attribute the largest value for the attribute plus one (1). That is, this key-code scans the specified attribute in the database for the largest number, and adds one (1). This effects attributes whose current value is zero (0), or **NULL**. If the attribute has a current value other than zero (0), or **NULL**, the command is ignored. This key-code operates for both numeric and character field types that contain numeric values.

As an example, assume there are 10 features in a theme which have an attribute called **ID**. The values of **ID** for these features begin at 1 and increase sequentially by 1, so that, the largest **ID** value is 10. Using the **ADDONE** key-code, the user is able to assign 11 as the default **ID** value for the next feature that is added to the theme.

The key-code **CURRENT_DATE** results in the **DEFAULT** command using as the default value for an attribute the current calendar date. The date string created with the **CURRENT_DATE** key-code is of the form *month/day/year*. For example the string *6/9/2004* would be created to represent June 9 2004.

If a custom date string is desired, the *m*, *mm*, *mmm*, *mmmm*, *d*, *dd*, *ddd*, *dddd*, *yy* and *yyyy* characters can be used to define a format that the date string is to

follow. These characters appear after the CURRENT_DATE option and are preceded with an underscore to indicate that a custom date format is being specified. For example:

CURRENT_DATE_mm/dd/yyyy

would result in the date string 06/09/2004 being created, while:

CURRENT_DATE_d/m/yyyy

would generate 9/6/2004 for June 9 2004.

If a space is to be inserted in the date format the underscore (_) character should be used. Spaces can not be used in specifying a custom date string. For example:

CURRENT_DATE_mmmm_d_yyyy

would create June 9 2004, while:

CURRENT_DATE_dd_mmmm_yyyy

would create 09 June 2004. Note that it is possible to include commas in the custom date string. For example:

CURRENT_DATE_dddd,_d_mmmm_yyyy

would generate Wednesday, 9 June 2004 for June 9 2004.

The m, d, and y characters represent the following:

- m month in numerical form with no leading zero
- mm month in numerical form with a leading zero, if need be
- mmm month in abbreviated text form (i.e. Sep for September)
- mmmm month in full text form (i.e. September, October, etc.)
- d day in numerical form with no leading zero
- dd day in numerical form with a leading zero, if need be
- ddd day in abbreviated text form (i.e. Wed for Wednesday)
- dddd day in full text form (i.e. Monday, Tuesday, etc.)

Table 2 - Database Design Type 2

DEFAULT	CURRENT_DATE_d/m/yyyy
DEFAULT	CURRENT_DATE_mm/dd/yyyy
DEFAULT	CURRENT_DATE_mmmm_d_yyyy
DEFAULT	CURRENT_DATE_dddd,_mmm_yyyy
DEFAULT	CURRENT_DATE_dddd,_d_mmmm_yyyy

- yy year in abbreviated form (i.e. 04 for 2004)
- yyyy year in full form (i.e. 2004)

- RETURN_YE y coordinate of a polyline/polygon feature end point

The order of how these characters are used in defining a date format string is completely up to the user. Some examples of custom date format strings are shown in Table 3.

The key-codes:

- RETURN_X,
- RETURN_Y,
- RETURN_XS,
- RETURN_YS,
- RETURN_XM,
- RETURN_YM,
- RETURN_XE and
- RETURN_YE

result in the **DEFAULT** command using as the default value for an attribute the x or y coordinate of a feature's keynode. Specifically, this includes:

- RETURN_X x coordinate of a point feature
- RETURN_Y y coordinate of a point feature
- RETURN_XS x coordinate of a polyline/polygon feature start point
- RETURN_YS y coordinate of a polyline/polygon feature start point
- RETURN_XM x coordinate of a polyline/polygon feature mid point
- RETURN_YM y coordinate of a polyline/polygon feature mid point
- RETURN_XE x coordinate of a polyline/polygon feature end point

The above key-codes can be enhanced to display latitude and longitude values rather than x,y coordinates by appending the _DD, _DMS, _LONG and _LAT strings. The _DD string denotes that latitude and longitude values are to be displayed in the form of decimal degrees. The _DMS extension indicates that the latitude and longitude values are to be displayed in the form of degrees, minutes and seconds, while the _LONG and _LAT indicate that longitude and latitude values are to be displayed in the form of degrees, minutes and seconds with direction (E or W for longitudes, N or S for latitudes), respectively.

In using the _DMS extension the default is to not include any digits to the right of the decimal point for the value of the seconds component. However, the user is able to override this by using the **SECONDS** command. This command enables the user to control the number of digits to the right of the decimal point that appears in the seconds component of a latitude or longitude value. This command should appear at the beginning of the *themes.txt* file and can appear in one of two forms:

SECONDS x,

where x is a number from 0 to 9, or

SECONDS ALL,

where ALL indicates, show as many digits as possible.

The default value, as mentioned above, is zero. In using the _LONG and _LAT

extensions, the convention is to not include any digits to the right of the decimal point for the value of the seconds component.

Command Option Number 3:

Below the **RANGE**, **CHOICES** or **DEFAULT** commands, the user can employ the **REMEMBER_LAST** command. The **REMEMBER_LAST** command indicates that the default or initial value of an attribute should be the last value that was used for the attribute. This command has an effect only when the value for an attribute is blank or NULL for an attribute of character type, or zero for an attribute of number type. This command is very useful when processing features that have not been assigned a value for an attribute and the user wishes to assign the value that was last specified for the attribute.

Note that it is possible for the **DEFAULT** and **REMEMBER_LAST** commands to both be assigned to an attribute. The reason being is when the software is first employed the last value assigned to an attribute does not exist. In this case, the **DEFAULT** value can be used. If the user has not specified a **DEFAULT** value, a blank string will be used as the default attribute value.

Once the user has assigned a value to an attribute, and the **REMEMBER_LAST** command has been applied to the attribute, the **DEFAULT** value will not be used whenever the value for an attribute is blank or NULL for an attribute of character type, or zero for an attribute of number type.

```
/*
SOURCE
SOURCE:
CHOICES
ATLAS
AS-BUILT
END
DEFAULT ATLAS
REMEMBER_LAST
```

In the example above, the default value that would be assigned to the **SOURCE** attribute is **ATLAS**. This will occur when a feature is added to a theme that is listed in the Theme Attribute Data File. Now, once a value for the **SOURCE** attribute has been specified, and if the user selects another feature whose value for the **SOURCE** attribute is **AS-BUILT**, and then selects a feature that has not been assigned a value for the **SOURCE** attribute. The default value for the **SOURCE** attribute for this feature would be **AS-BUILT** and not **ATLAS**, because the **REMEMBER_LAST** command has been applied to the **SOURCE** attribute.

Command Option Number 4:

Below the **RANGE**, **CHOICES**, **DEFAULT** or **REMEMBER_LAST** commands, the user has the option of using, the **ATTRIBUTE_EQUATION** command. The **ATTRIBUTE_EQUATION** command indicates that the value of an attribute is based upon a user-defined equation and not on direct user entry. That is to say, this command populates the value of an attribute by solving an equation that the user specifies.

Below the **ATTRIBUTE_EQUATION** command, on the very next data line, the user specifies the desired equation. Note that a comment line can not appear between the **ATTRIBUTE_EQUATION** command line and the equation line. The equation must appear on the data line below the **ATTRIBUTE_EQUATION** command line. Furthermore, the entire equation must appear on a single data line, and can not be split up amongst several data lines.

Within an equation, the user is able to:

- access the values stored in other attributes,
- apply mathematical operators and/or functions, and
- specify conditionals. As an example, the following could appear in the *Theme Attribute Data File*:

```
/*
SZ
Starting Elevation:
/*
EZ
Ending Elevation:
/*
LEN
Line Length:
/*
SLOPE
Line Slope (%):
ATTRIBUTE_EQUATION
((EZ-SZ)/LEN)*100.0
```

In the example above there are four attributes (**SZ**, **EZ**, **LEN** and **SLOPE**). The values assigned to the **SZ**, **EZ** and **LEN** attributes will be those that the user explicitly enters. The value assigned to the **SLOPE** attribute, however, will be determined by solving the $((EZ-SZ)/LEN)*100.0$ equation and *not by user input*. The values to be used in the above equation are those currently assigned to the **SZ**, **EZ**, and **LEN** attributes, and as displayed in the multi-input dialog box. Should the user hit the enter key following the entry of one of the **SZ**, **EZ** or **LEN** attributes, the value for the **SLOPE** attribute would be updated, in the multi-input dialog box, to reflect the entry.

Equations are solved proceeding left to right resolving expressions within parentheses first. When parentheses are nested they are resolved inside to outside. That is, the inner most set of parentheses are solved first, then the next inner most and so forth. When there are more than one nest of parentheses, each nest is resolved inside to outside and from left to right. The user is able to use parentheses to ensure that the desired sequence of calculations is performed.

The supported Mathematical operators include (see note 1 below):

- \wedge = Exponentiation,
- $/$ = Division,
- $*$ = Multiplication,
- $+$ = Addition,
- $-$ = Subtraction.

The supported Boolean operators include:

1. = = Equality,
2. < = Less than,
3. > = Greater than,
4. ≤ = Less than or equal to,
5. ≥ = Greater than or equal to,
6. ◇ = Not equal to.

The supported Functions include:

ABS	= absolute value of a number,
ACOS	= arc cosine of a number expressed in radians (see note 2 below),
ASIN	= arc sine of a number expressed in radians (see note 2 below),
ATAN	= arc tangent of a number expressed in radians (see note 2 below),
COS	= cosine of a number expressed in decimal degrees (see note 3 below),
DEG2RAD	= conversion of degrees to radians,
RAD2DEG	= conversion of radians to degrees,
ROUND	= rounding up of a number to the nearest whole number,
SIN	= sine of a number expressed in decimal degrees (see note 3),
SQRT	= square root of a number greater than zero,
TAN	= tangent of a number expressed in decimal degrees (see note 3),
TRUNCATE	= truncation of a number.
STRING	= treat the value as a string, not as a numeric value.
TRIM	= remove leading and trailing blanks from the value.

Note 1: Mathematical operators appearing in an equation are resolved by evaluating exponentiation first, division and multiplication second, and addition and subtraction last. Division and multiplication operations are resolved proceeding left to right in the order they appear. Likewise, addition and subtraction operations are resolved in a similar manner.

Note 2: The input values for the arc sine (ASIN), arc cosine (ACOS) and arc tangent (ATAN) functions are expressed in radians, while the values they generate will be expressed in decimal degrees.

Note 3: The input values used with the sine (SIN), cosine (COS) and tangent (TAN) functions should be expressed in decimal degrees, not radians. The values returned by these functions, however, will be in terms of radians.

Note 4: The STRING function is used when the result of an equation is to be a string and not a number. Since it is possible for numbers to be stored in fields (attributes) that are of string type, as well as number type, the default is to treat all attribute values as numbers. If the values of a field (attribute) are to be treated as strings, the STRING function must be used.

Note 5: If the result of an equation is to be of string type the only supported mathematical operator is addition (+). In this case the individual arguments are directly concatenated. Should the user wish to enter a string explicitly, the double-quote character, “, must enclose the string. For example if the equation “A” + “B” appeared, the

result would be AB. No space would be inserted between the arguments. If a space (blank character) is to separate the values, the equation should appear as: “A” + “ + ”B”.

The user also has the ability to specify a conditional clause by utilizing the IF operator. In so doing, the user can introduce some decision making capabilities into the attribute value calculation. The IF operator follows the following syntax:

IF(expression,true_condition,false_condition)

In using the IF operator the user specifies an *expression* that will be evaluated such that it results in either a *true* or *false* condition. As such, the user must specify the *true* and *false* conditions, in addition to the *expression*.

For example, if an attribute A2 was assigned the equation IF(Q2=4.5,0.5,0.75) the command would assign A2 the value 0.5, if the value for the attribute Q2 was equal to 4.5, or the value 0.75, if the value for the attribute Q2 was not equal to 4.5. **Note** that if the *true* or *false* condition is to contain an equation, **the user must** enclose the condition in parentheses in order for it to be properly evaluated (refer to the, (Q2^0.092*1.0808), portion of the sample equation 5 below).

Shown in Table 4 is the correct syntax for various sample equations. Note that J2, K2, P2, R2, Q2, M2, L2, S2, N2, PIN1, PIN2, PIN3, MAP, SLN, BLDNUM, HOUSENUM, ST_DIR and STREET represent sample attributes (fields).

It is noted that in the sample equations there are no leading, trailing or embedded blank spaces. However, such blank spaces may be introduced if deemed necessary for the sake of clarity. If so, they will be disregarded by the program.

Table 4 - Sample Equations

1. J2-K2
2. $29.8 * P2 * R2 * Q2^2 * \text{SQRT}(M2)$
3. $((J2-20) / L2)^{0.54 * N2}$
4. S2*N2
5. $\text{IF}(Q2=2.5, 0.9, \text{IF}(Q2=4.5, 0.78, \text{IF}(Q2=1.9, 0.97, \text{IF}(Q2=2.5, (Q2^{-0.092} * 1.0808), 4.0))))$
6. $\text{DEG2RAD}(\text{ACOS}(0.5))$
7. $\text{ASIN}(0.75)$
8. $\text{ATAN}(0.35)$
9. $\text{SIN}(30)$
10. $\text{COS}(60)$
11. $\text{TAN}(45)$
12. $\text{ROUND}(4.75)$
13. $\text{TRUNCATE}(4.75)$
14. $\text{STRING}(\text{PIN1}) + \text{STRING}(\text{PIN2})$
15. $\text{STRING}(\text{PIN1}) + "-" + \text{STRING}(\text{PIN2}) + "-" + \text{STRING}(\text{PIN3})$
16. $"\text{My String} = " + \text{STRING}(\text{PIN1}) + "000-" + \text{STRING}(\text{PIN2}) + "0010"$
17. $\text{IF}(\text{MAP}=\text{SLN}, "\text{TRUE_CONDITION}", "\text{FALSE_CONDITION}')$
18. $\text{IF}(\text{BLDNUM} <> " ", (\text{TRIM}(\text{HOUSENUM}) + \text{TRIM}(\text{ST_DIR}) + \text{TRIM}(\text{STREET})), \text{BLDNUM})$
19. $\text{TRIM}(\text{HOUSENUM}) + \text{TRIM}(\text{ST_DIR}) + \text{TRIM}(\text{STREET})$

If you have a request for Command Of The Month, feel free to phone, fax or e-mail your request to The CEDRA Corporation.