McIDAS-X Learning Guide

Version 2020

McIDAS-X Learning Guide Version 2020 Table of Contents

McIDAS Learning Guide - Introduction

This Learning Guide introduces you to McIDAS and teaches you the basic tools and concepts needed to use the system. The Learning Guide assumes you know meteorology; however you do not need to know McIDAS.

The Learning Guide consists of the eleven lessons in the Table of Contents. The lessons build upon one another so you will learn most effectively by completing them in order. The Learning Guide is applicable for users running McIDAS-X. To complete the lessons in this Learning Guide, you will access the real time data from a server maintained by Unidata.

The original McIDAS-X Learning Guide, which was published by SSEC in September 1994, was converted to HTML format by Unidata in September 1996. Most recently, the HTML format was updated in 2020 by SSEC to include information about changes through the McIDAS-X version 2020.1 release (see the Revision History for more details).

What is McIDAS?

It's the Man computer Interactive Data Access System

Overview: McIDAS is a suite of applications for analyzing and displaying meteorological data for research and education. McIDAS has been in use and under continual development by the University of Wisconsin-Madison Space Science and Engineering Center (SSEC) since 1972. The software can be used with conventional observational, satellite, and grid-point data. McIDAS-X is available for use on Red Hat Linux, Apple and Microsoft platforms (currently supported platforms).

Features:

- *Multiple data types.* McIDAS may be used to analyze the full range of meteorological data: conventional, gridded numerical forecast, radar, and satellite data.
- *Access to satellite imagery.* The McIDAS software can be used to analyze any of the major satellite data sets: GOES, MSG, HIMAWARI, POES, JPSS, MODIS, FY-2, etc.
- *Access to NEXRAD imagery.* The McIDAS software can be used to analyze NEXRAD products.
- *Standard meteorological analysis programs.* The software can perform standard meteorological analyses for conventional observation and grid-point data.
- *Fully navigated displays.* All earth-depicting displays produced by McIDAS are fully navigated; earth positioning is available for every point on the screen.
- *Multiple visualizations.* Observational data can be analyzed into grid-point data, and grid-point data can be transformed into image visualizations.
- *Multiple graphic overlays.*
- *Remapping of images into other coordinate projections.*
- *Multiple frames and windows.* McIDAS has multiple frames for image and graphic displays and multiple windows for text; the number of frames is defined by the user and limited only by workstation resources.
- *Image and graphic storage.* McIDAS can store images and graphics in internal memory for quick redisplay and for combining displays into an animation.
- *Scheduled operations.* McIDAS includes a scheduler facility that allows sites to create analyses at user-specified times.
- *Tools for site-specific software development.*

Getting Started

Table of Contents:

Getting Started

The **M**an-**c**omputer **I**nteractive **D**ata **A**ccess **S**ystem (McIDAS) is a powerful data ingest, management, and analysis system for meteorological research and operational weather forecasting. The McIDAS database contains both real-time weather satellite and conventional weather data which you can access and analyze. For example, with McIDAS you can:

- view a time sequence of real-time satellite images
- track cloud motions
- color enhance displayed images
- overlay surface and upper air contoured analyses on satellite imagery
- calculate derived parameters
- display thermodynamic diagrams

This Learning Guide introduces you to McIDAS-X and teaches you the basic tools and concepts needed to use the system. The lessons build upon one another, so you will learn most effectively completing them in order. This lesson describes how to:

- start and exit McIDAS
- log on to the workstation
- use online helps
- raise windows
- stop commands

The following commands are used in this lesson:

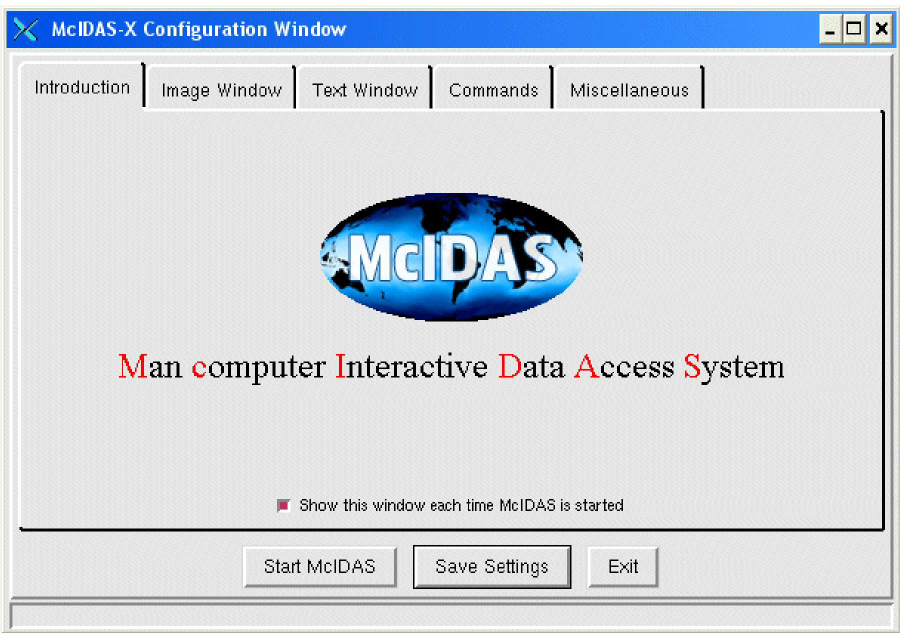
Command	Function
EXIT	exits the McIDAS session
GLOBKEY	lists global keyword parameters and examples
HELP	lists online documentation
KILL	stops a command
LOGON	logs you on to the McIDAS system
+	raises the image window
?	lists the currently running McIDAS commands
/	stops a command

Getting Started - Basic Concepts

Configuration GUI

Before you begin using this Learning Guide, McIDAS must be installed on your workstation and the account you are using must be enabled to run McIDAS. The first time you run McIDAS-X, the `$HOME/.mcidasrc` configuration file is created and the McIDAS-X Configuration Graphical User Interface (GUI) is run. The `.mcidasrc` file stores flags that configure the McIDAS session to your preferences. As shown below, the configuration GUI contains several tabs with sliders, checkboxes and text fields that are used to select values that override the settings of the corresponding flags in the `.mcidasrc` file. This configuration GUI makes one-time (for the current session only) configuration changes easier and less error-prone than other methods like changing flag settings in the `.mcidasrc` file or specifying the flags in the command line. By changing some of the start-up flags in the `$HOME/.mcidasrc` file, you can:

- set the number of frames
- allocate additional frame memory
- run commands at startup
- set the text fonts
- set the command and output recall
- set the workstation display
- set the McIDAS Text and Command Window options
- set the McIDAS Image Window options



The **Start McIDAS** button starts a McIDAS session with the current settings as defined by the Configuration GUI.

The **Save Settings** button saves the current GUI settings in the `.mcidasrc` file. To make recovery easier, the option first moves the current `.mcidasrc` to `.mcidasrc.old` before creating the new `.mcidasrc` with the GUI settings.

The **Exit** button exits the McIDAS configuration window without saving any changes or starting McIDAS.

The Configuration GUI tabs

- **Introduction**

  The Introduction tab displays a title page, as well as an option to display the configuration GUI each time McIDAS is started.

- **Image Window**

  The Image Window tab contains an option to add, modify, and delete frame specifications. The scrollbars allow the user to change the horizontal and vertical positioning of the image window and to edit the number of image and graphic colors used by McIDAS. There are also options to automatically resize the image window, and use 8-bit PseudoColor visual mode.

- **Text Window**

  The Text Window tab contains scrollbars to change the width and height of the text window, as well as the horizontal and vertical position. There are also options to change the number of scroll lines in the text window, and the number of commands to be saved in the command history. The drop down menus allow the user to change the look of the text window with font and color options.

- **Commands**

  The Commands tab contains an option to add, modify, and delete start-up commands that will be executed automatically when McIDAS starts. Any McIDAS command may be included in this list.

- **Miscellaneous**

  The Miscellaneous tab contains a variety of checkboxes, including options to start the GUI upon McIDAS startup, enable the McIDAS scheduler, enable character case inversion, specify a working directory, and to use the X Windows graphical pixel map to draw the cursor. There is also a scrollbar to change the amount of allocated memory for additional frames, and a memory efficiency optimization option.

Getting Started - Basic Concepts (Continued)

McIDAS Windows

The McIDAS-X session generates two windows:

- image window
- text and command window

Windows can be *active* and *raised*. Active windows normally receive all keyboard input. To make a window *active*, move the pointer onto the title bar and click the left mouse button. By activating a window, you also *raise* it. Raising a window brings it to the top of any overlapping windows.

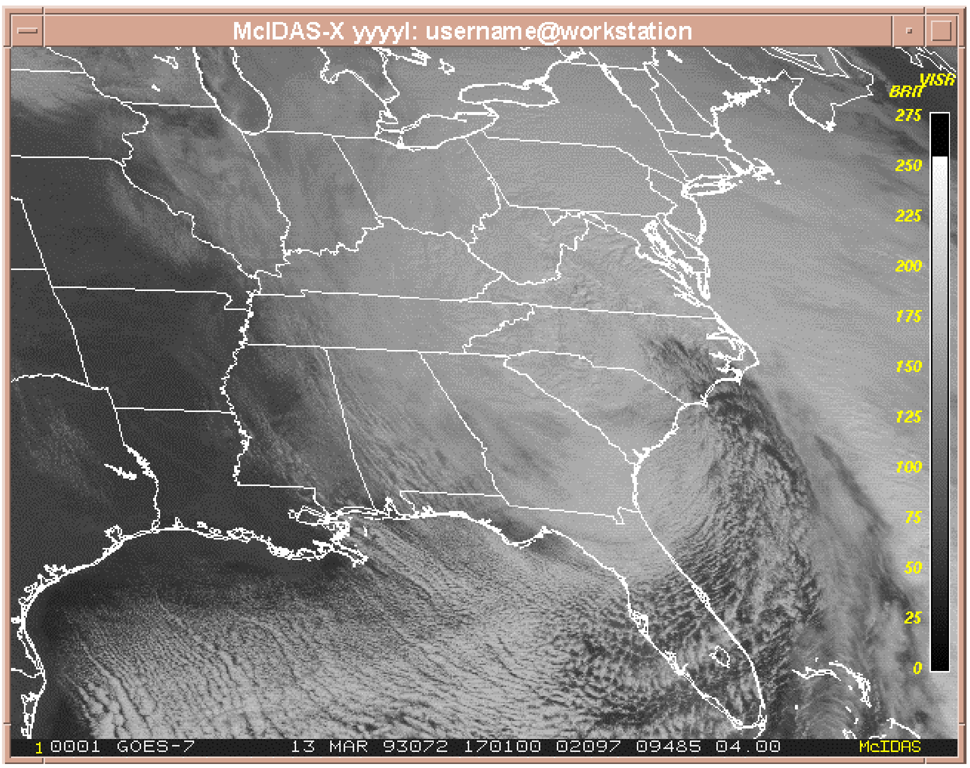
All McIDAS-X windows have a title bar that contains the following information:

> **McIDAS-X** ***yyyyl*****:** *username@workstation*

where *l* is the fastrack version if applicable and *username@workstation* displays your user name and the workstation running your McIDAS session.

Image Window

The image window shows frames that display McIDAS images and graphics. An image is a pictorial representation of data, for example, a satellite image. A frame can display an image, graphic, or both, like the one shown below.

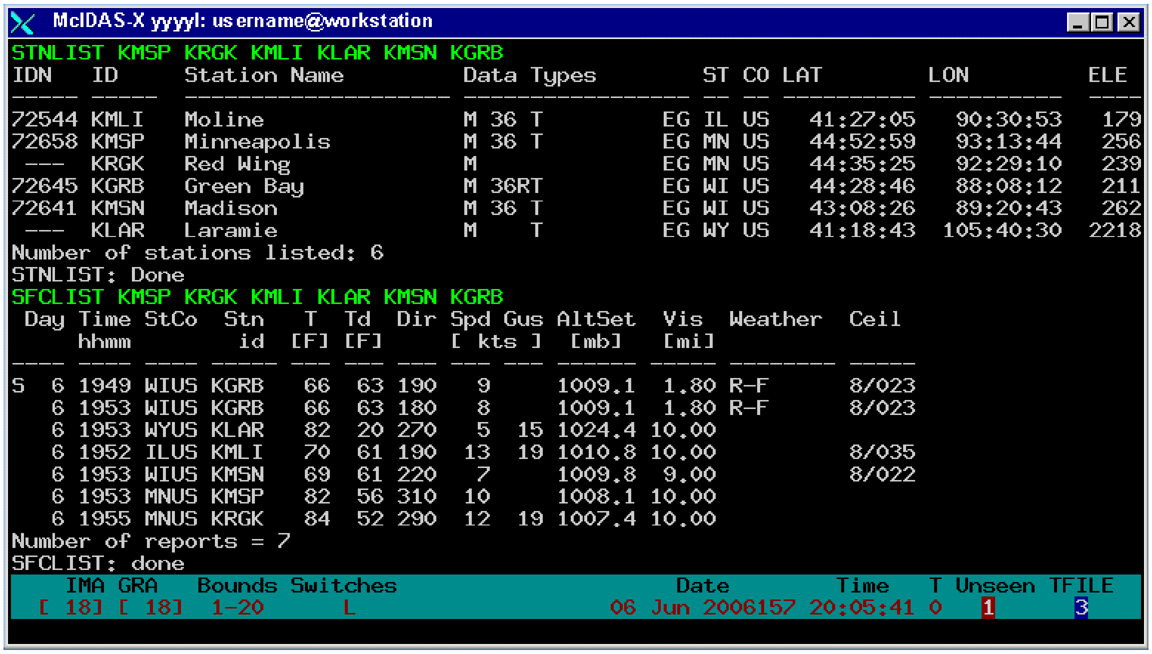


Text and Command Window

Use the text and command window to enter McIDAS commands, to view the status of your McIDAS session and to display textual information from McIDAS commands. A status line appears in the lower portion of the window above the command line. It contains information about the current looping status, loop bounds, current image frame number, date, time and text frame status. There are ten text frames within each McIDAS text and command window that can be used to display textual information such as helps for McIDAS commands, data information, and a history of executed commands.

You can scroll in a text frame using the arrow keys or the PgUp/PgDn keys on the keyboard. You can switch between text frames with the numeric keypad on your keyboard.

The examples below show the text output from the STNLIST and SFCLIST commands in the text and command window.



Workstation Status

Each text and command window has a status line at the bottom which shows some information about the state of your McIDAS session. The status line from above is shown and explained below.

McIDAS-X status line:

```
IMA GRA Bounds Switches Date Time T Unseen TFILE [18] [18] 1-20 L 06 Feb 2006157 20:05:41 0 1 3
```

The McIDAS-X status line above lists the current image frame (18), the current graphics frame (18), the loop bounds (1-20), command switches (L - looping turned on), current date in UTC (Gregorian [06 Jun] and Julian [2006157]), the current UTC time (20:05:41) and the current text frame (0). Unseen indicates that text has been written to the text frame indicated in red (1) since the last time it was opened. TFILE indicates that the output in the highlighted text frame listed below it in blue (3) is being written to a file with the TFILE command. If there are no Unseen windows, or TFILE is not running, these labels will not appear on the line.

Three-button Mouse

You can use either a two or three-button mouse with McIDAS, but a three-button mouse is preferred. In all cases, when you are instructed to use the right mouse button, use the rightmost button of your mouse and when instructed to use the left button, use the leftmost button. When instructed to use the middle mouse button, use the middle button on a 3-button mouse or click both buttons simultaneously on your 2-button mouse.

Getting Started - Basic Concepts (Continued)

Commands

McIDAS applications are command-driven, which makes the system very flexible. There are two basic command formats in McIDAS:

- single letter commands
- multiple letter commands with positional parameters, keywords, and quoted text

Positional parameters are words, numbers, or letters that further define a command. You must enter positional parameters in the exact order specified in the command format. It is not always necessary to use every positional parameter. If you want to specify the default value for a positional parameter, enter an **X** in its place.

You can also add *keywords* to the command to further clarify a command function. Keywords are optional and their order in the command is not important, provided they follow the positional parameters and precede quoted text.

Some arguments (positional parameters, keyword parameters) like dates, latitudes, and longitudes must conform to certain formats. The ARGHELP command can be used to get a listing of these argument formats.

Most keywords are command specific; however the seven global keywords below can be used with any McIDAS command.

- **DEV=**
- **FONT=**
- **MCCompress=**
- **MCStretch=**
- **PAN=**
- **TCOL=**
- **TWIN=**
- **VIRT=**

The DEV= keyword specifies the destination device of text output generated by a command. The FONT= keyword specifies the font for drawing text on the image window. The MCCompress= keyword specifies the data compression method the ADDE remote server uses when sending requested data to the workstation. The PAN= keyword specifies the frame panel number for displaying output or reading frame information. The TCOL= keyword specifies the text output color for text written to the text window. The TWIN= keyword specifies the destination text frame for text message output. The VIRT= keyword specifies a virtual graphics number to write the graphics output to. See the McIDAS User's Guide or run the GLOBKEY command for more information.

A sample command line is shown below:

ZA	**3 20 C**	**POS=200 200**	**"McIDAS**
|	\ /	|	|
command	***parameters***	***keyword***	***quoted text***

To run single letter commands, simultaneously press the Alt key and the letter key, or type the letter and press Enter. Below is an example of the format for single letter commands in the Learning Guide.

Press: **Alt A**

To run multiple letter commands, type the command and press Enter. This Learning Guide assumes you will always press Enter after typing the command. Below is an example of the command format in the Learning Guide.

Type: **CW 1 3**

This means you should type **CW 1 3** and then press Enter. You can have multiple spaces between parameters and keywords.

A command line can contain several commands. To concatenate commands, type a semicolon between the commands. For example:

Type: **IMGDISP MYDATA/IMAGES.8000 3;LS 1-3**

When multiple commands are entered on the command line, each command is run in the order it is listed. The first command must finish before the next command can begin.

Some commands have a quote (") field that indicates all text following the quote belongs to that command. When concatenating commands with quote field, you need to replace the double quote with a left brace ({) and place a right brace before the semicolon to surround the quote field. For example, typing:

SKE X 00:00 1 "SFCLIST KMSN 12;ERASE F 1

would enter the commands **SFCLIST KMSN 12** and **ERASE F 1** into the command scheduler file to run at a later time (00 UTC in this example). But, typing the following:

SKE X 00:00 1 {SFCLIST KMSN 12};ERASE F 1

would enter the command **SFCLIST KMSN 12** into the command scheduler file to run at 00 UTC, but run **ERASE F 1** immediately to erase the graphics on frame 1.

Several keys are available for editing the command line. Listed below are command line editing keys and their functions.

Key	**Function**
Home	moves the cursor to the beginning of the line
End	moves the cursor to the end of the line
Insert	toggles the insert typing mode
Delete	deletes the character over the cursor
Backspace	deletes the character to the left of the cursor
Right/Left arrows	moves the cursor one character to the right or left
Tab, Shift Tab	moves the cursor one parameter to the right or left
Enter	runs the command
Esc	erases the command from the command line and places the cursor at the beginning of the line
Alt ?	displays an abbreviated help for the current command in the text frame
& and ^	recalls a command if the command line is empty; & recalls the previous command from a list of the last 100 commands entered;^ scans the call list in the opposite direction
Ctrl LeftMouseButton	prints the latitude and longitude of the cursor position on the command line

Command Stack

Command Mode-entered commands are maintained in a list that can be scanned forward or backward. In McIDAS-X, the number of commands to remember is specified in the `.mcidasrc` file, with a default of 1000. If the command line is empty, you can scan backward through the list of commands using the ampersand (**&**) key and scan forward through the list using the caret (^) key. These recalled commands can be reviewed, re-executed, or edited and then executed. This feature is very useful when learning to use commands. You can easily add and/or change parameters and keywords on previously executed commands without having to retype the entire command.

Command Helps

The online helps provide an abbreviated description of the McIDAS commands. For a complete description of each command, see the McIDAS User's Guide.

You can also get information on the status of your session using the F command. The output includes the number and size of frames, current and opposite frame numbers, current loop bounds, K and L toggle states, cursor type, cursor size, cursor color and its position on the frame as shown below.

```
Video Status for Your Workstation Frames(s) --------- Number Available 6 Current 1 (Opp = 4 ) Loop Bounds 1 to 6 Visible
(K & W toggle) Yes / Yes Looping (L toggle) No Cursor parameters: Size = 31 / 31 Type = Xhair Center position = 206 / 396
Color = RED Image frames 1 - 6 with imbedded graphics are 480 BY 640
```

Getting Started - Starting and Exiting McIDAS

To start or stop a McIDAS session, follow the steps below:

1. Start McIDAS by entering the following in lowercase from the Unix prompt.
   Type: `mcidas or mcidas -config`

   If this is the first time that you've run McIDAS, the `mcidas` command will display the McIDAS Configuration GUI.

2. Change the Configuration GUI so that your options match those listed below:
   - 6 image frames - each 480 lines by 640 elements
     Click: the **Image Window** tab
     Click: **10 @ 480 x 640**
     Click: **Modify**
     Change the number of frames to 6
   - 2 MB memory allocated for creating additional image frames
     Click: the **Miscellaneous** tab
     Change the amount of memory allocated to 2 MB

   On the **Introduction** tab, click the **Show this window each time McIDAS is started** button to uncheck the box. The next time you start McIDAS, the McIDAS Configuration GUI will not show up. In the future, you can also display this GUI by typing `mcidas -config`. You can also change the display by editing the *flags* in the `$HOME/.mcidasrc` file by using a unix editor. For more information on the start-up flags, read the documentation within the `.mcidasrc` configuration file.

3. Save your changes and exit the configuration GUI.
   Click: **Save Settings**
   A text warning will pop up telling you that a new version of `.mcidasrc` will be created with the current settings.
   Click: **OK**
   Click: **Exit**
4. Start McIDAS again. This time you will notice that mcidas started automatically without the configuration GUI.
   Type: `mcidas`

   When the McIDAS GUI starts, a text warning will pop up telling you that the GUI is going to analyze your Server List and that it would take a while. Click "OK". A LOGON screen will then appear. You can just click "Cancel". We will update the GUI tables in the Graphical User Interface lesson.

5. Exit and restart McIDAS one last time with the configuration GUI to turn off the option to start the main GUI each time McIDAS is started. We will get back to the GUI in the Graphical User Interface lesson. Additionally, recheck the **Show this window each time McIDAS is started** box. The configuration GUI is a useful tool that will assist you in starting each McIDAS session as you go through the Learning Guide.
   Type: **EXIT**
   Type: `mcidas -config`

   Under the **Introduction** tab, click the **Show this window each time McIDAS is started** box, and under the **Miscellaneous** tab, click the **Start GUI upon McIDAS startup** box to deselect this option. Save your new settings and start McIDAS.

Getting Started - Logging on to the Workstation

You can start McIDAS without logging on to the workstation as you did at the beginning of this lesson; however, some system and data files (e.g. grid files and string tables) you create or edit won't have your initials or project number saved with them. The LOGON command with the **I** (initialization) parameter modifies the session by:

- clearing the string table
- setting the image frame loop bounds from one through one-half the total number of frames
- setting the default dwell rate to nine units for the first frame and six units for the remaining frames
- setting the graphics line width to one pixel
- initializing the graphics color levels
- displaying a red, 31 by 31 pixel, cross hair cursor
- setting the enhancement table to 0 255 0 255 0 255 0 255
- clearing the buffer of remembered commands

1. Log on and initialize the workstation. (If you don't have your own initials and project number, use **DEMO** for *initials* and **1234** for *project*.)
   Type: **LOGON** *initials project* **I**

   The message "LOGON to McIDAS-X completed" is displayed in the active text window.

   The LOGOFF command can be used if you are leaving your workstation and don't want others to use your initials and project number to access data.

Getting Started - Using Online Helps

Online helps list the syntax of each command including the parameters, keywords, and remarks. To access the online help, type HELP in the McIDAS text and command window followed by the command for which you'd like more information. You can find additional command information in the McIDAS User's Guide.

1. Find the help for the ZLM command.
   Type: **HELP ZLM**

   The help text appears in the active text frame as shown below:

```
ZLM -- Draws graphics using the mouse ZLM BOX color height width ZLM CIRCLE color height width ZLM DRAW color
SMOOTH= ZLM FILL color height width ZLM FREE color Parameters: BOX | draws a box CIRCLE | draws a circle or ellipse
DRAW | draws a smooth, continuous line as you move the mouse FILL | draws and fills a box FREE | draws straight
lines between points selected with mouse (def) color | graphics color level, use 0 to erase with FILL option (def=3)
height | height of the box, circle or ellipse, in pixels (def=cursor height) width | width of the box, circle or
ellipse, in pixels (def=cursor width) Keywords: DASh=YES | draw dashed lines for BOX, CIRCLE, DRAW and FREE options
=NO | solid lines are drawn (def) GRA= | graphics frame number to display output; valid only with MODE=N and BOX,
CIRCLE, or FILL option (def=current) MODe=I | interactive mode (def) =N | non-interactive "batch" mode; draws shape
centered at line and element position specified in POS keyword on the graphics frame specified with the GRA keyword;
valid only with BOX, CIRCLE, or FILL option POS=line ele | TV line and element position to center the shape on;
valid only with MODE=N and BOX, CIRCLE, or FILL option (def=current cursor position) SMOoth= | number of samples to
take for the running average with the DRAW option; the range is 1-50; larger numbers produce smoother lines (def=3)
----------
```

2. Get a listing of all the commands and a brief description of their functions.
   Type: **HELP**
3. If you are entering a command and have forgotten the syntax, you can get a brief listing of the syntax by pressing Alt ?. Show the syntax of the IMGDISP command.
   Type: **IMGDISP** (but don't press Enter)
   Press: **Alt ?**

   The syntax of the IMGDISP command is listed in the current text frame.

4. Use the Esc key to clear the line.
   Press: **Esc**

Getting Started - Raising Windows and Displaying Text Frames

Raising a window brings it to the front of any overlapping windows so you can view the entire window.

Use the Text and Command Window for entering McIDAS commands, displaying text output, and displaying workstation status information. You can reposition the Text and Command Window to a different location on the display, and you can also resize it.

The Text and Command Window displays keyboard input regardless of whether it or the Image Window is active. Click in either window to make it active.

Use the numeric keypad's plus(+) key to make the Image Window active. To make the Text and Command Window active, press a numeric keypad number from 0-9. The number you press determines which text frame is displayed in the window. 0 is the default text window displayed when McIDAS is first ran.

1. To raise the Text and Command Window and display text frame 1,
   Press: **1**
2. To raise the Image Window,
   Press: +
3. Change the Text and Command Window back to text frame 0.
   Press: **0**

Getting Started - Stopping Commands

Every command being run has a Process IDentification (PID) number and a Parent Process IDentification number (PPID) associated with it. To stop a command on the workstation, you must find the PID for the command by checking the command status with the question mark (?) command and then using the KILL command. In this exercise, you will enter a command and then stop it.

1. Enter the following command.
   Type: **ZLM**
2. Find the PID for the ZLM command by checking the command status.
   Press: **?**

```
UID PID PPID C STIME TTY TIME CMD user 593 362 0 08:59:33 ? 0:00 /usr/dt/bin/dtexec -open 0 -ttprocid > user 564 563
0 08:59:26 ? 0:01 xterm -title zeppo : Xterm -e ksh -c > user 7870 7869 0 14:28:37 pts/13 0:02 mcimage -imageColors
128 -autoResize > user 565 564 0 08:59:26 pts/12 0:00 ksh -c echo Trying to establish conne> user 568 565 0 08:59:27
pts/12 0:00 rlogin zeppo user 574 569 0 08:59:28 pts/13 0:01 -csh user 7869 7863 1 14:28:36 pts/13 0:01 mctext -iw
-c !@exec mcimage -imageCo> user 601 598 0 08:59:35 pts/14 0:00 rlogin zeppo -l mcidas user 7863 1 0 14:28:20 pts/13
0:01 mcenv -k 7849 -f 16 -e 5m -f 4@480x64> user 673 374 1 09:01:59 pts/3 1:32 netscape -install user 7921 1 1
14:50:36 pts/13 0:00 ZLM
```

3. Find the PID for the ZLM command. Type KILL followed by the appropriate PID number and press Enter. For example, to stop PID 7921:
   WARNING: Be sure that you get the correct PID number. Accidently stopping the wrong process will produce unpredictable results.
   Type: **KILL 7921**
   Alternate method (use the '/' (slash) command):
   Type: **/ 7921**
4. Check the command status again to make sure ZLM is no longer running.
   Press: **?**
   You should see that ZLM is no longer listed in the list of processes.

5. You can now EXIT McIDAS since we are at the end of this lesson.
   Type: **EXIT**

ADDE

Table of Contents:

ADDE

This section of your *McIDAS Learning Guide* contains information about the ADDE (Abstract Data Distribution Environment) software in the core McIDAS package. The ADDE allows your workstation to act as a client, efficiently accessing data from multiple McIDAS-X servers. This chapter introduces you to ADDE concepts and use.

McIDAS is in transition from the use of non-ADDE programs to ADDE replacements. Most of the examples in this Learning Guide have been updated to use ADDE versions of commands. For those functions where an ADDE replacement is not available, the non-ADDE commands are used.

This introduction provides an overview of the ADDE system. The following topics are included:

- definitions of commonly used terms
- an explanation of client and server functions
- the scheme used for naming data
- information about look-up tables
- information about naming data on your local server
- using compressed data transfers

The following commands are used in this lesson.

Command	Function
AKA	manages and lists aliases for dataset names
DATALOC	manages and lists the client routing table
DSINFO	lists datasets on local and remote servers
DSSERVE	manages and lists dataset names on the local server
IMGCOPY	copies image data from one dataset to another
IMGLIST	lists images in a dataset

ADDE - Terminology

The terms below are used throughout this chapter. Remember, because a workstation can store and display data, it can act as both a server and a client.

Term	Definition
Alias	a short, user-defined name representing an ADDE dataset name; for example, the alias GV1 might represent the dataset name SSEC-RT/GOES8-1KVIS
Client	workstation receiving and displaying data
Dataset	collection of one or more files with a common format; for example, one dataset may consist of areas 101 to 104, another dataset may consist of MD files 1 to 10
Dataset Name	name used by ADDE to reference a dataset; the dataset name consists of a group name and a descriptor name separated by a slash (/); for example, SSEC-RT/GOES8-1KVIS
Descriptor	name used to reference a dataset; for example, a dataset of areas 105 to 108 containing GOES-7 visible data at 4 km resolution might have the descriptor G7-VIS-4K
File Number	number used to reference a McIDAS format file (the standard McIDAS numbering scheme)
Format	file format; McIDAS formats include area, grid and MD; non-McIDAS formats include HDF and netCDF
Group	name used to reference a collection of descriptors; used by the client to determine which server should receive data requests
Position	absolute or time-relative position of a file in a dataset; position numbers greater than zero represent an absolute position in the dataset; position numbers less than or equal to zero represent a relative position, 0 is the most recent and -1 is the next most recent; for example, if a dataset has four images with times 13 UTC, 14 UTC, 15 UTC and 12 UTC, they have the positions -2, -1, 0 and -3
Server	machine storing and supplying data
Type	data type: image, grid, point, navigation, or text

ADDE - Clients and Servers

ADDE distributes data using networked servers and clients. Servers store data and distribute it to the client. Clients request and receive data and run applications that use the data. Clients and servers communicate using the TCP/IP communications protocol.

Each account running McIDAS-X acts as both a client and local server. When a client requests data from the local server, it searches for the data in the directories specified in the user's **`MCPATH`** environment variable, another directory specified in a REDIRECT entry or a directory in the DIRFILE keyword of DSSERVE.

The client can also request data from a remote server. A remote server can be any of the following:

- a different account on the same McIDAS-X workstation configured as a remote server
- another McIDAS-X workstation configured as a remote workstation

The difference between a local server and a remote server on a McIDAS-X workstation is that the data stored in a local server is available only to the McIDAS-X sessions started under that account name. Data in a remote server is accessible to all McIDAS ADDE clients. Remote servers are discussed more thoroughly in the *Site Administration and Management* section of the ADDE portion of the McIDAS User's Guide.

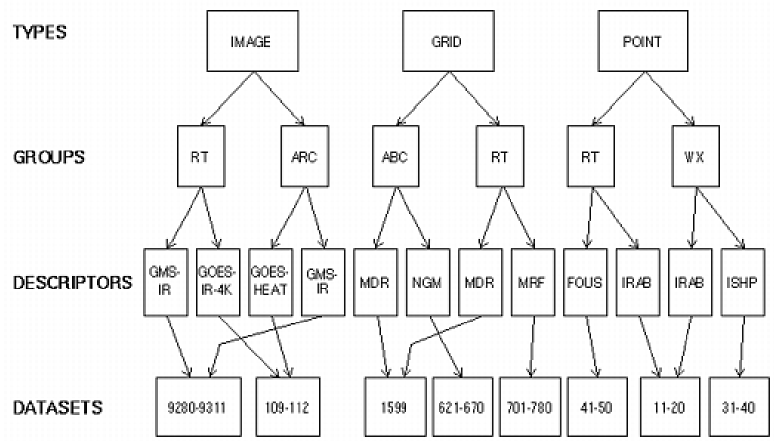
ADDE - Data Naming Scheme

In the non-ADDE core commands, all image, grid and point files are referenced by file numbers. If you don't know the file numbers, finding data can be difficult. The ADDE commands use dataset names (in a group/descriptor format) that map to datasets. Groups and descriptors can be any name that the user chooses. If the descriptors and group names follow a logical convention, it's easy to locate the data.

The naming scheme for datasets consists of three parts:

- type
- group
- descriptor

Figure 1. Data Naming Scheme



Type is the top tier in ADDE's hierarchical naming scheme. Image, grid and point indicate the type of data in the dataset files. McIDAS area files are image data; McIDAS grid files are grid data; McIDAS MD files are point data.

Group is the next tier in the naming scheme. A group name can be used only once under each type. The group tier in Figure 1 has three groups named RT. Notice that RT is used only once under each type.

Descriptors are the bottom tier in the naming scheme. Descriptors are not data files; they are names that point to datasets. Identical descriptors under different groups can point to the same or different datasets.

Since each ADDE command that accesses data works only with a single type, you don't have to specify the type in the command entry. For example, the command IMGDISP does not have a parameter or keyword for type because it limits its search to datasets of type IMAGE. The following command displays the most recent image in the dataset named SSEC-RT/G16-IR-B13 on frame 1, centered on St. Louis, Missouri.

IMGDISP SSEC-RT/G16-IR-B13 1 STATION=STL

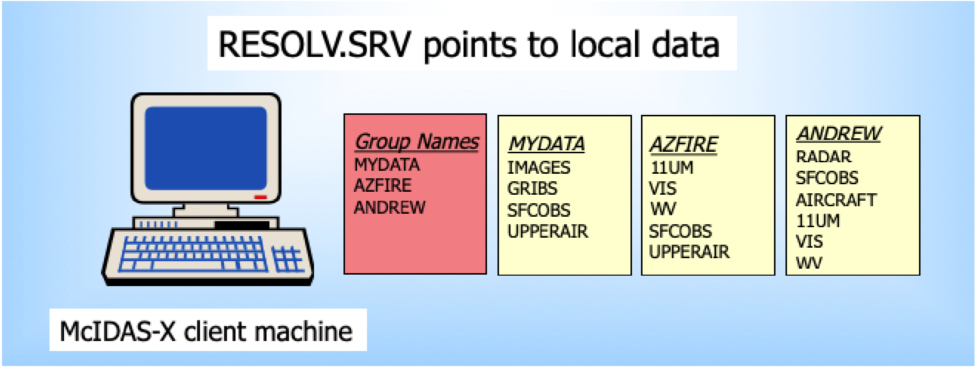
ADDE - Naming Datasets for ADDE Local and Remote Access

Use the DSSERVE command to create dataset names on your local server. DSSERVE assigns the group and descriptor to a dataset consisting of one or more files of the specified format. You can also add a description of up to 80 characters. For example, the following command assigns the dataset name HURR/IMAGES to areas 7001 through 7199 on the local server:

DSSERVE ADD ANDREW/VIS AREA 7001 7199 "Vis images of Hurricane Andrew

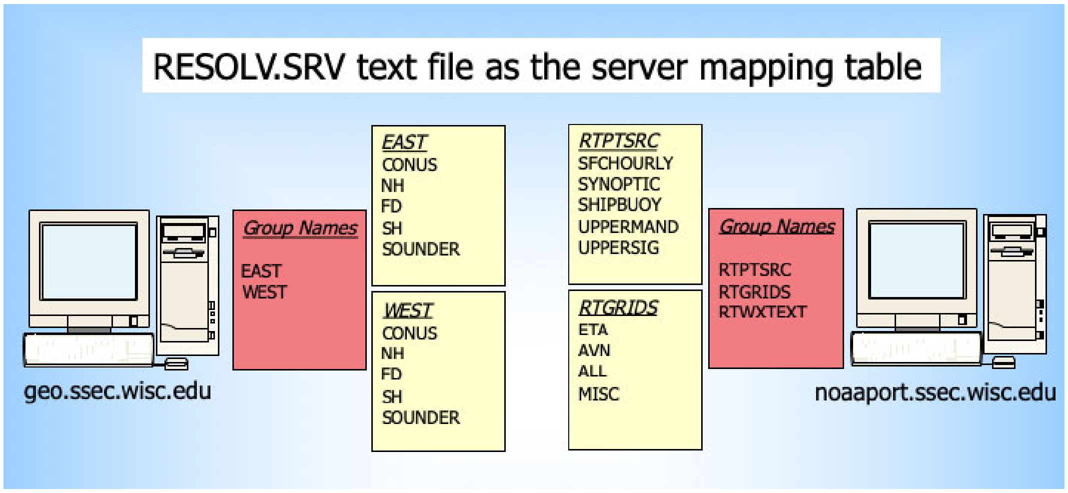
You can use the same group or descriptor with each data type. For example, if you have image, grid and point data for a case study of a major flood event, you may want to use the group name FLOOD for all the datasets. Or if you have large numbers of images that you use as part of a demonstration, you may want to use the group name DEMO with logical descriptor names like TSTORM, FOREST-FIRES, ECLIPSE and HURR-HUGO. DSSERVE writes to the RESOLV.SRV file which is created locally on the user's machine to access data from a user's machine, directory mount, network attached storage (NAS), etc. *Figure 1* contains a graphical view of this.

Figure 1. When datasets are created with the DSSERVE command, the entries are saved in RESOLV.SRV (e.g. Datasets in the MYDATA group are MYDATA/IMAGES, MYDATA/GRIBS, MYDATA/SFCOBS, and MYDATA/UPPERAIR).



Use the DSSERVE command to list or modify the server mapping table. Clients can only modify their local server's mapping table; they cannot modify a remote server's mapping table. The remote server's administrator will similarly use the DSSERVE command to add datasets for clients to access. *See Figure 2.*

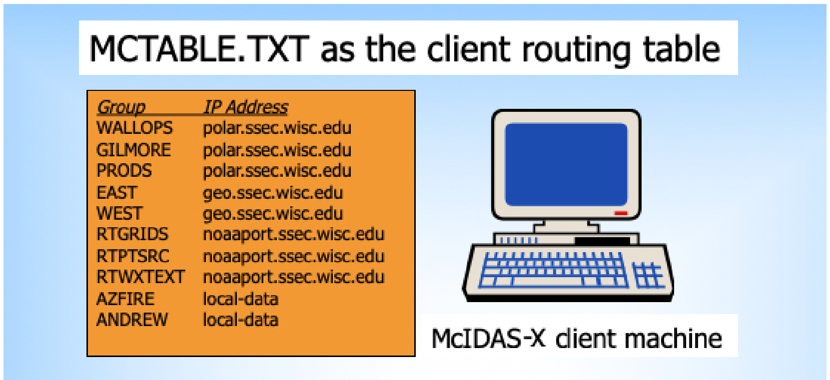
Figure 2. When datasets are created with the DSSERVE command on the server, the entries are also saved in RESOLV.SRV which functions as the server mapping table.



ADDE - Accessing Local and Remote Data

Each ADDE client command starts a transaction by requesting data from a server. Both the client and server must recognize the command's specified dataset name in their look-up tables. On the client, this table is called the *routing table* because it determines which server to route the data request to. The client routing table contains two lists: a list of group names with their associated server IP addresses, and a list of aliases with the dataset names they represent. See *Figure 1.*

Figure 1. When groups are assigned server IP addresses with the DATALOC command, the entries are saved in MCTABLE.TXT.



Use the DATALOC command to manage the list of group names. Use the AKA command to manage the list of aliases. Aliases allow you to use a short, easily remembered name to represent a dataset name in other ADDE commands. You can use this alias regardless of which server the dataset is stored on. For example, you could create the alias G16FD to represent the dataset name SSEC-RT/G16-FULLDISK.

Each user can access multiple client routing tables. This optional feature is useful when a site table has the default routing information and alias names for all users, and each user has a local table with individualized routing information and alias names. The default file for your local table is **`MCTABLE.TXT`**. The site table, if it exists, is maintained by your site administrator or operations staff and is usually write-protected so you can't modify it. Use the DATALOC and AKA commands to list information from all the client routing tables accessible to you.

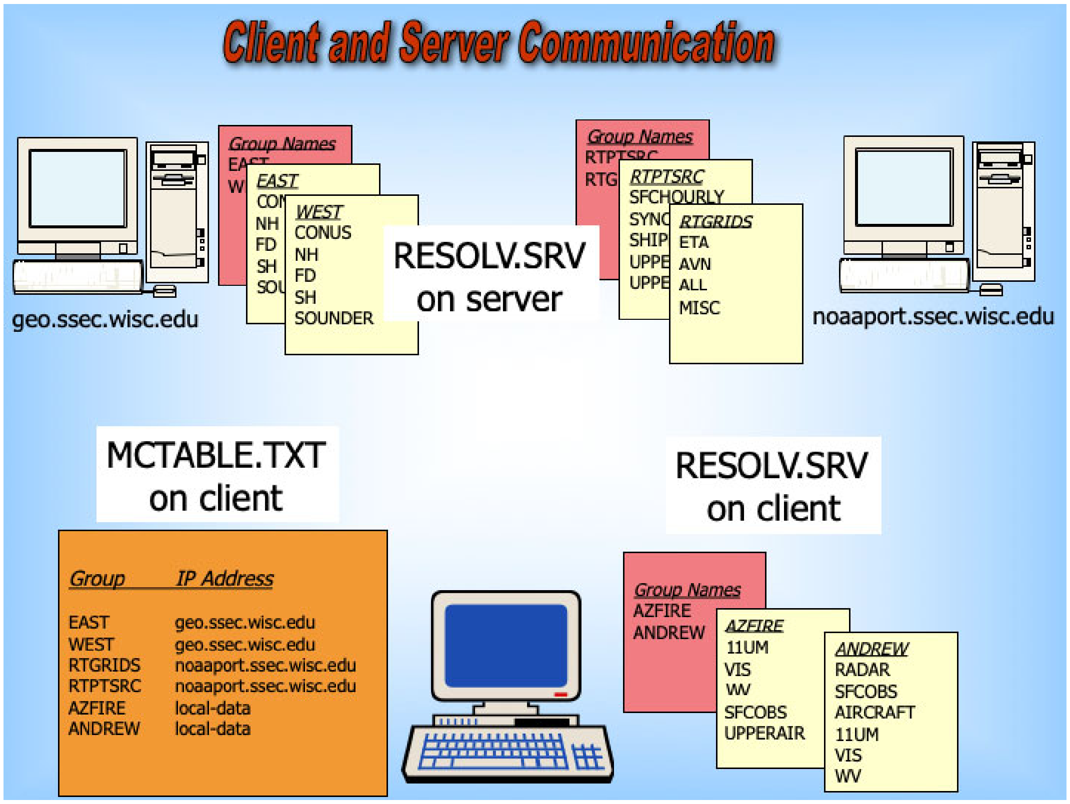
If you decide to use multiple client routing tables, you must set the Unix environment variables **`MCTABLE_READ`** and **`MCTABLE_WRITE`**. See the McIDAS User's Guide for more information on using multiple routing tables.

When you enter an ADDE command, the client routing table is scanned for an entry with the specified group name or alias. If an entry is found, the data request is routed to the server specified in the entry. If the group name or alias is not found in the client routing table, the request is routed to the local server.

When naming data on your local server, you must use group names that are not mapped to other servers in your client routing table. Use the command entry DATALOC LIST to list the group names in your client routing table. Since ADDE commands that request data from a group name not found in the client routing table are automatically routed to the local server, you do not have to add your local server's group names to the table.

To summarize, the server mapping table maps dataset names to the files that make up the datasets. On the server, this table is called the *mapping table* because it maps the group and descriptor to a specific image, grid, point, navigation or text dataset. When the server receives a data request from a client, it reads the mapping table to locate the correct dataset. The server sends the requested data back to the client so it can list or display the data. See *Figure 2.*

Figure 2. An overview of how the client and server system files work together to serve and display data.


Client and Server Communication
Group Names
EAST
CONUS
NH
FD
SH
WEST
CONUS
NH
FD
SH
SOUNDER
RESOLV.SRV
on server
geo.ssec.wisc.edu
Group Names
RTPTSRC
RTPTSRC
SFCHOURLY
RTGRIDS
ETA
AVN
ALL
MISC
noaaport.ssec.wisc.edu
MCTABLE.TXT
on client
RESOLV.SRV
on client
Group IP Address
EAST geo.ssec.wisc.edu
WEST geo.ssec.wisc.edu
RTGRIDS noaaport.ssec.wisc.edu
RTPTSRC noaaport.ssec.wisc.edu
AZFIRE local-data
ANDREW local-data
Group Names
AZFIRE
ANDREW
AZFIRE
11UM
VIS
WV
SFCOBS
UPPERAIR
ANDREW
RADAR
SFCOBS
AIRCRAFT
11UM
VIS
WV

ADDE - Using Compressed Data Transfers

There are three ADDE data transfer methods available for sending data from remote servers to your client workstation. The methods are gzip-compressed, compressed, and uncompressed.

The gzip-compressed and compressed methods are most beneficial to users who transfer large data files or those with slow data connections. However, they only reduce the amount of data during the transfer; they do not change the resultant file size or accounting (the byte counts remain the same). For example, the destination file size from an IMGCOPY command is the same no matter which of the three methods you use. If most of the datasets you access are on a local network with fast transfer rates, then using the uncompressed method is probably best because it minimizes overhead.

The data transfer method you use on your client workstation is determined by the setting of its **`MCCOMPRESS`** environment variable. Because it's an environment variable, the method you select is the default for the entire McIDAS-X session. The only way to change the default method is to exit your McIDAS-X session, change the setting of **`MCCOMPRESS`**, then restart McIDAS-X. To override the default setting on a command-by-command basis, use the **`MCCOMPRESS`** global keyword with any ADDE command.

See the McIDAS User's Guide for more information on using compressed data transfers.

ADDE - Getting Started

The exercise below introduces you to ADDE concepts such as dataset names, aliases, clients and servers. Use this exercise to practice adding entries to your client routing table and local server's mapping table. The image listing, display and copy commands will be explained further in the Satellite Imagery Section.

1. Start a McIDAS session.
   At the Unix prompt:
   Type: **`mcidas -config`**
   Your session should still be set for six frames from the last time you changed the settings in the Configuration GUI. Set it for twenty frames, save the settings as described in Getting Started.
2. List your client routing table's groups and associated server IP addresses.
   Type: **DATALOC**

   If this is the first time you are using the ADDE, the table is empty and the following listing is displayed.

   ```
   Group Name Server IP Address -------------------- ----------------------------------------
   ```
3. Modify your client routing table so that commands requesting data from a real-time dataset containing free data from Unidata. The group RTGOESR is routed to the Unidata remote server with the IP address ADDE.UCAR.EDU.
   Type: **DATALOC ADD RTGOESR ADDE.UCAR.EDU**
4. List all datasets in the group RTGOESR.
   Type: **DSINFO ALL RTGOESR**
5. List all datasets of type IMAGE in the group RTGOESR.
   Type: **DSINFO I RTGOESR**

   Note that instead of the full word, 'IMAGE,' only an **I** is used. Each parameter in DSINFO can be abbreviated with its first letter instead of writing it out in its entirety.

   The first part of the listing is displayed below. The NumPos value is the number of positions in the dataset. When setting up servers with newer data types, NumPos of 99999 usually means that the server can access an unlimited number of files on the remote machine.

   ```
   Dataset Names of Type:  IMAGE in Group:  RTGOESR Name NumPos Content ------------ ------
   ---------------------------------------- CONUS 99999 GOES-East CONUS all bands CONUSC01 99999 GOES-East CONUS 0.47 um
   VIS aerosol-over-land CONUSC02 99999 GOES-East CONUS 0.64 um VIS clouds fog/insol/winds CONUSC03 99999 GOES-East
   CONUS 0.86 um Near IR veg/burn scar/aerosol CONUSC04 99999 GOES-East CONUS 1.37 um Near IR cirrus cloud CONUSC05
   99999 GOES-East CONUS 1.61 um Near IR cloud phase/snow CONUSC06 99999 GOES-East CONUS 2.24 um Near IR land/cloud
   vege/snow CONUSC07 99999 GOES-East CONUS 3.89 um IR Sfc/cloud/fog/fire/winds
   ```
6. List the 11 most recent images in the dataset RTGOESR/CONUSC13 in reverse chronological order.
   Type: **IMGLIST RTGOESR/CONUSC13.-10**

   The following listing is a sample of what is displayed. The Pos value is the absolute position of the image in the dataset. The image listed first is the most recent image, relative position 0. The next image is the second most recent image, relative position -1...

   ```
   Image file directory listing for:RTGOESR/CONUSC13 Pos Satellite/ Date Time Center Band(s) sensor Lat Lon ---
   ------------- ------------ -------- ---- ---- ------------ 576 GOES-16 16 SEP 19259 02:16:14 30 87 13 575 GOES-16 16
   SEP 19259 02:11:14 30 87 13 574 GOES-16 16 SEP 19259 02:06:14 30 87 13
   ```
7. List the two most recent images (positions 0 and -1) in the dataset RTGOESR/FD (all bands are included in this dataset) with expanded information on the available Band (1).
   Type: **IMGLIST RTGOESR/FD.-1 BAND=1**

   The following listing is an example of what is displayed. The Pos value is the absolute position of the image in the dataset. The Res values are the resolution of the images' center pixels.

   ```
   Image file directory listing for:RTGOESR/FD Pos Satellite/ Date Time Center Res (km) Image_Size sensor Lat Lon Lat
   Lon --- ------------- ------------ -------- ---- ---- ----- ----- ------------ 192 GOES-16 16 SEP 19259 02:10:17 0
   75 Band: 1 0.47 um VIS aerosol-over-land 1.00 1.00 10848 x10848 191 GOES-16 16 SEP 19259 02:00:17 0 75 Band: 1 0.47
   um VIS aerosol-over-land 1.00 1.00 10848 x10848 IMGLIST: done
   ```
8. Create the aliases G16C and G16FD for the dataset names RTGOESR/CONUS and RTGOESR/FD.
   Type: **AKA ADD G16C RTGOESR/CONUS**
   Type: **AKA ADD G16FD RTGOESR/FD**
9. List the aliases in your client routing table.
   Type: **AKA**

   The following listing is displayed.

   ```
   Alias Name Group/Descriptor -------------------- ---------------------------------------- G16C RTGOESR/CONUS G16FD
   RTGOESR/FD
   ```
10. Display the most recent image in the dataset with the alias name G16C. Display the image on frame 1, band 13, centered on Athens, Georgia, and draw a map on it. The AKA listing in step 9 shows that the alias G16C represents dataset RTGOESR/CONUS.

Type: **IMGDISP G16C 1 BAND=13 STATION=KAHN SF=YES;MAP VH 2**

11. Assign the dataset name MYDATA/IMAGES to all of the areas available on your local workstation.
    Type: **DSSERVE ADD MYDATA/IMAGES AREA 1 9999 "ALL AREA FILES**
12. Use the IMGLIST command to find five empty areas on the local server.
    Type: **IMGLIST MYDATA/IMAGES.4000 4004**

    If data exists in this block of areas, move or delete the areas or use the IMGLIST command to find an empty block of 5 areas.
13. Assign the dataset name MYDATA/TEST-IMAGES to the empty block of areas.
    Type: **DSSERVE ADD MYDATA/TEST-IMAGES AREA 4000 4004 "Scratch areas for testing**

    A listing similar to the following is displayed.

    ```
    Group/Descriptor Type Format & Range RT Comment ----------------------- ----- ----------------- -- ------------------- MYDATA/IMAGES IMAGE AREA 1-9999 ALL AREA FILES MYDATA/TEST-IMAGES IMAGE AREA 4000-4004 Scratch areas for testing
    ```
14. List the groups in your client routing table.
    Type: **DATALOC**

    A listing similar to the following is displayed.

    ```
    Group Name Server IP Address -------------------- --------------------------------------- RTGOESR ADDE.UCAR.EDU MYDATA <LOCAL-DATA> <LOCAL-DATA> indicates that data will be accessed from the local data directory.
    ```
15. Create the alias TI for the dataset name MYDATA/TEST-IMAGES.
    Type: **AKA ADD TI MYDATA/TEST-IMAGES**
16. List the aliases in your client routing table.
    Type: **AKA**

    The following listing is displayed.

    ```
    Alias Name Group/Descriptor -------------------- --------------------------------------- G16C RTGOESR/CONUS G16FD RTGOESR/FD TI MYDATA/TEST-IMAGES
    ```
17. List all the images in the dataset with the alias name TI. The AKA listing in step 16 shows that the alias TI represents dataset MYDATA/TEST-IMAGES.
    Type: **IMGLIST TI.ALL**

    The following output is displayed because the dataset has no images.

    ```
    Image file directory listing for:TI IMGLIST: No images satisfy the selection criteria
    ```
18. Copy the most recent image in the dataset with the alias name G16C to position 5 in the dataset with the alias name TI. Copy band 13 and place Athens, Georgia, at its center. The AKA listing in step 16 shows that the aliases G16C and TI represent datasets RTGOESR/CONUS and MYDATA/TEST-IMAGES.
    Type: **IMGCOPY G16C TI.5 STATION=KAHN BAND=13**
19. List all the images in the dataset with the alias name TI.
    Type: **IMGLIST TI.ALL**

    An example of the listing is displayed.

    ```
    Image file directory listing for:TI Pos Satellite/ Date Time Center Band(s) sensor Lat Lon --- ------------- ------------ -------- ---- ---- ------------ 5 GOES-16 16 SEP 19259 02:10:17 34 83 13
    ```
20. In addition to the MYDATA/TEST-IMAGES local *area* file dataset, you can setup local datasets as defined in the DSSERVE help section. In the following exercises you will access GOES-R Series netCDF format files with the IMG* commands by using DSSERVE to assign a dataset name to the files. When doing so, specify "ABIN" in the *format* parameter, TYPE=IMAGE, and the directory and file masks in the DIRFILE keyword. The files in this exercise are mission-standard Level 1b or Level 2 files in netCDF-4 format, like those from NOAA CLASS or from the GRB data stream after they've been decoded with CSPP Geo or other software. The names of files should look similar to ***ABI-L1b-RadC-M3C01_G16_s2015229195720.nc*** (current CSPP Geo naming convention) or ***OR_ABI-L1b-RadC-M4C16_G16_s20151702215532_e20151702220362_c20151702220394.nc*** (GOES-R Ground System naming convention).

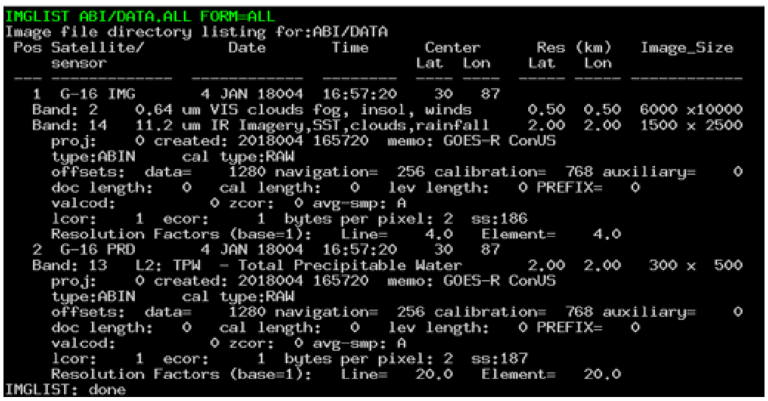
    Use ftp via web browser or command line instructions below to download the following three files:

    OR_ABI-L1b-RadC-M3C02_G16_s20180041657204_e20180041659577_c20180041700012.nc
    OR_ABI-L1b-RadC-M3C14_G16_s20180041657204_e20180041659577_c20180041700023.nc
    OR_ABI-L2-TPWC-M3_G16_s20180041657204_e20180041659577_c20180041701161.nc

    ftp ftp.ssec.wisc.edu
    login: anonymous
    password: your email

binary
prompt
cd pub/mug/Data/ABI/NetCDF
mget *.nc
exit

21. Create a local dataset with DSSERVE to access the 2 *RadC* ABI netCDF files on your local machine in the directory, ***<local-path>/Data/ABI/NetCDF***.
    Type: **DSSERVE ADD ABI/CONUS ABIN TYPE=IMAGE DIRFILE='<local-path>/Data/ABI/NetCDF/*RadC*'**
22. List the most recent image in the dataset (the default is position 0, so it doesn't need to be specified).
    Type: **IMGLIST ABI/CONUS**
23. Datasets can be organized however the user chooses. Make a dataset that includes all of the files in the data directory.
    Type: **DSSERVE ADD ABI/DATA ABIN TYPE=IMAGE DIRFILE= '<local-path>/Data/ABI/NetCDF/*'**
24. IMGLIST can list the image directory and band information with the FORM=ALL keyword. Your IMGLIST command should match the image.
    Type: **IMGLIST ABI/DATA.ALL FORM=ALL**



```
IMGLIST ABI/DATA.ALL FORM=ALL
Image file directory listing for:ABI/DATA
 Pos Satellite/         Date       Time      Center      Res (km)   Image_Size
     sensor                                  Lat  Lon    Lat   Lon
 --- -------------  ------------  --------  ---- ----  ----- ----- ------------
   1  G-16 IMG       4 JAN 18004  16:57:20    30   87
   Band: 2    0.64 um VIS clouds fog, insol, winds      0.50  0.50  6000 x10000
   Band: 14   11.2 um IR Imagery,SST,clouds,rainfall    2.00  2.00  1500 x 2500
     proj:    0 created: 2018004 165720  memo: GOES-R ConUS
     type:ABIN     cal type:RAW
     offsets:  data=    1280 navigation=  256 calibration=  768 auxiliary=    0
     doc length:   0   cal length:   0   lev length:   0 PREFIX=   0
     valcod:           0 zcor:  0 avg-smp: A
     lcor:    1  ecor:     1  bytes per pixel: 2  ss:186
     Resolution Factors (base=1):   Line=    4.0   Element=    4.0
   2  G-16 PRD       4 JAN 18004  16:57:20    30   87
   Band: 13   L2: TPW  - Total Precipitable Water       2.00  2.00   300 x  500
     proj:    0 created: 2018004 165720  memo: GOES-R ConUS
     type:ABIN     cal type:RAW
     offsets:  data=    1280 navigation=  256 calibration=  768 auxiliary=    0
     doc length:   0   cal length:   0   lev length:   0 PREFIX=   0
     valcod:           0 zcor:  0 avg-smp: A
     lcor:    1  ecor:     1  bytes per pixel: 2  ss:187
     Resolution Factors (base=1):   Line=   20.0   Element=   20.0
IMGLIST: done
```

25. Another IMGLIST keyword, FORM=BAND, lists the band and resolution without the file’s expanded listing.
    Type: **IMGLIST ABI/DATA.1 BAND=2 FORM=BAND**
26. Exit McIDAS.
    Type: **EXIT**

Loop Control System

Table of Contents:

Loop Control System

The loop control system is a group of McIDAS commands for viewing image frames in a sequence. The order of frames in a loop can be sequential or random. For example, when image frames are loaded with a chronological series of satellite images, you can loop them to follow cloud motions. This lesson describes how to:

- create a loop
- change the dwell rate
- change the loop sequence
- add image frames to your McIDAS session

The following commands are used in this lesson.

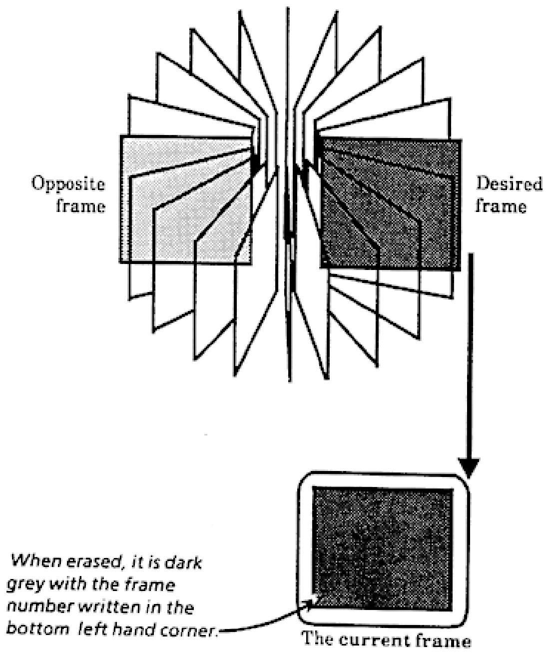
Command	Function
A	advances one frame
B	backs up one frame
DR	defines the dwell rate
F	displays the workstation state
L	turns looping on and off
LB	sets frame loop bounds
LS	defines a looping sequence
MAKFRM	adds new frames to a McIDAS session
O	displays the opposite frame

Loop Control System - Basic Concepts

You can define a random or sequential loop of image frames, and set the display time (dwell rate) for each frame in the loop.

Loop Bounds and Sequences

Image frames in the image window are like slides in a slide projector carousel, in that any frame can be brought easily into view or looked at in any order, but you cannot look at more than one at the same time in the same image window:



A loop has beginning and ending frames called loop bounds. When you start McIDAS, the loop bounds are set as frames one through the total number of frames. If you log on to McIDAS and initialize the workstation, the loop bounds are one through one-half the total number of image frames. You will use the commands LB and LS to set loop sequences which can be sequential or random.

When you define loop bounds, you also create opposite loop bounds. Each frame has an opposite frame that is half the total number of frames away. For example, on a session with ten frames, frames 1 and 6 are opposite, frames 2 and 7 are opposite, etc. The single letter command O toggles between a frame and its opposite, which is useful for comparing images.

Dwell Rate

The dwell rate is the amount of time each frame is displayed during automatic looping. The default dwell rate is nine units for the first frame and six units for the remaining frames. In McIDAS-X, one unit is appoximately 1/15th of a second; the fastest dwell rate is 1/15 of a second (1 unit), though some workstations

Loop Control System - Creating a Loop

In this exercise, you will start a McIDAS session with twenty frames, display six images with the IMGDISP command and loop the frames using Alt L. Then, you will switch to the opposite loop with the Alt O command.

1. Start a McIDAS session.
   At the Unix prompt:
   Type: **`mcidas`**
   Your session should still be set for twenty frames from the last time you made changes in the Configuration GUI. If not, set it for twenty frames, as described in Getting Started.
2. Check the loop bounds.
   Press: **Alt F**
   The default loop bounds are 1 through 20 as shown. Notice that the loop bounds in the status line is *1-20*.
   ```
   Video Status for Your Workstation Frame(s) -------- Number of Available 20 Current 1 (Opp = 11 ) Loop Bounds 1 to 20 Visible (K & W toggle) Yes/Yes Looping (L toggle) No Cursor parameters: Size = 31 / 31 Type = Xhair Center position = 10 / 10 Color = RED Image frames 1 - 20 with imbedded graphics are 480 BY 640
   ```
3. Display six images using the following IMGDISP commands. IMGDISP is discussed further in the Satellite Imagery lesson.
   Type: **IMGDISP G16C.2 1 STATION=KDCA BAND=8 REPEAT=3**
   Type: **IMGDISP G16C.2 4 STATION=KDCA BAND=13 REPEAT=3**
4. Set the loop bounds with the LS command, then loop frames 1 through 3.
   Press: **LS 1-3;LS O 4-6**
   Press: **Alt L**
   Notice that the status line at the bottom of the Text and Command window changes to show an L under Switches and the Frame number changes as the loop progresses:
   ```
   IMA GRA Bounds Switches Date Time T [2] [2] 1-3 L 16 Sep 2019259 02:43:31 0
   ```
5. Toggle to the opposite loop.
   Press: **Alt O**
   The loop displays frames 4 through 6 and the bounds in the status line changes to 4-6.

6. Stop the loop.
   Press: **Alt L**
   The loop display returns to the first frame in the current loop sequence, in this case, frame 4. Notice that the status line now has a value of 4 in the Frame position and the L is not listed in the switches section.
   ```
   IMA GRA Bounds Switches Date Time T 4 4 4-6 16 Sep 2019259 02:44:01 0
   ```
7. Toggle to the primary loop.
   Press: **Alt O**
8. Use the single letter commands A and B to manually step through the current loop. Alt A advances through the frames; Alt B moves backwards through the frames.
   Press: **Alt A**
   Press: **Alt B**

Loop Control System - Changing the Dwell Rate

In this exercise, you will use the DR command to change the dwell rate, which is the amount of time each frame is displayed during automatic looping. Each frame can have a unique dwell rate assigned by DR.

1. Check the dwell rates.
   Type: **DR**
   The default dwell rate is nine units for the first frame and six units for the remaining frames.

2. Start the loop.
   Press: **Alt L**
3. Change the dwell rate to two units.
   Type: **DR 2**
   The images continue looping, but at a faster rate. Each frame in the loop is displayed for two units.

4. Change the dwell rate to 20 units.
   Type: **DR 20**
   Each frame in the loop bounds is displayed for 20 units.

5. Set the dwell rate to a variety of values.
   Type: **DR 2 10 20**
   The first frame is set to two units, the second frame to 10 units, and the third frame to 20 units.

6. Stop the loop.
   Press: **Alt L**
7. Change the dwell rate to its initialized value.
   Type: **DR INI**
   The dwell rate should be nine units for the first frame and six units for the remaining frames.

Loop Control System - Changing the Loop Sequence

In this exercise, you will change the loop sequence with the LB and LS commands.

1. Change the loop bounds to be from frames 1 through 6.
   Type: **LB 1 6**
   Notice that the loop bounds on the status line changes to *1-6* and frame 1 is displayed.

2. Start and stop the loop.
   Press: **Alt L**
   Press: **Alt L**
3. Define a loop with random sequencing.
   Type: **LS 1 3 2**
   The primary loop is frames 1, 3 and 2, and the opposite loop is frames 11, 13, and 12. Notice that the status display at the bottom of the text and command window changes to indicate that the loop is random.
   ```
   IMA GRA Bounds Switches Date Time T 1 1 random 16 Sep 2019259 03:02:35 0
   ```
4. Set the opposite loop to display a random sequence.
   Type: **LS O 6 5 4**
5. Check the current loop settings.
   Type: **LS**
   The primary loop is frames 1, 3, and 2, and the opposite loop is frames 6, 5, and 4. The dwell rate for both loops is nine units for the first frame and six units for the remaining frames.

6. Loop the primary sequence, the loop set in step 3.
   Press: **Alt L**
   Notice the frames loop from frame 1 to 3 to 2.

7. Toggle to the opposite loop sequence.
   Press: **Alt O**
8. Toggle back to the primary loop sequence.
   Press: **Alt O**
9. Stop looping.
   Press: **Alt L**

Loop Control System - Adding Image Frames

You can create additional image frames after McIDAS has been started using the MAKFRM command. New image frames are added to the end of the current frame list. The default size of the new frame is the same as that of the frame currently displayed.

1. Add two new frames the same size as the current frame (480 x 640) to your McIDAS session.
   Type: **MAKFRM 2**
   Two frames are added to the session, giving you twenty-two frames.

2. Check the number of image frames for this session.
   Press: **Alt F**
   Status information for your workstation is displayed as shown below. The number of available frames is twenty-two.

```
Video Status for Your Workstation Frame(s) -------- Number of Available 22 Current 1 (Opp = 11 ) Loop Bounds 1 to 20
Visible (K & W toggle) Yes/Yes Looping (L toggle) No Cursor parameters: Size = 31 / 31 Type = Xhair Center position
= 10 / 10 Color = RED Image frames 1 - 22 with imbedded graphics are 480 BY 640
```

3. Exit McIDAS.
   Type: **EXIT**

Satellite Imagery

Table of Contents:

Satellite Imagery

McIDAS sites can receive real-time weather satellite data that you can easily access and analyze using McIDAS-X. This lesson describes how to:

- display satellite data
- use coordinate systems
- list image data
- change the image resolution
- copy and display images
- manipulate image data
- delete images
- use briefing windows

The following commands are used in this lesson.

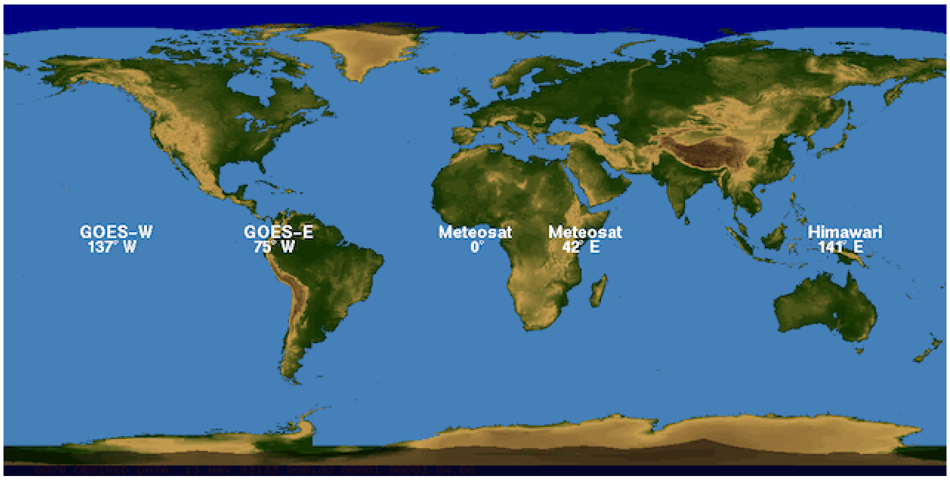
Command	Function
CUR	defines cursor size, type and color
D	lists the image data at the cursor center
E	lists the earth coordinates at the cursor center
FRMLABEL	displays an image label on a frame
FRMLIST	lists a frame directory
IMGCOPY	copies images
IMGDEL	deletes images
IMGDISP	displays an image on a frame
IMGLIST	lists image directories
IMGOPER	applies mathematical functions to image data
IMGPROBE	lists or plots image data for a region of the displayed image
IMGREMAP	remaps images into different projections
MAP	displays a map and latitude/longitude lines on a graphics frame
PC	positions the cursor at a desired point on the frame
SYNC	synchronizes loops running in multiple briefing windows

Satellite Imagery - Basic Concepts

SSEC receives real-time satellite images from geostationary and polar orbiting satellites. Geostationary satellites remain above a fixed location on the earth's surface, appoximately 35,800 km above the equator. Because the satellites rotate with the earth, they always observe the same portion of the globe. Typically, there are operational geostationary satellites from the United States, the European Organisation for the Exploitation of Meteorological Satellites (EUMETSAT), and the Japan Meteorological Agency (JMA).

At most times, the United States has a satellite called GOES-East, which monitors North and South America and the western Atlantic Ocean, and one called GOES-West, which monitors North and South America and the eastern Pacific Ocean. EUMETSAT has two satellites (METEOSAT) which monitor Europe, Africa, Asia, the Indian Ocean, and the Atlantic Ocean. JMA has a satellite (Himawari) which monitors eastern Asia, Australia, the western Pacific Ocean and the eastern Indian Ocean.

Roll your mouse over the image below to see the coverage of each geostationary satellite:



Polar orbiting satellites orbit at much lower altitudes (800-900 km). Their path is 2,400 km wide centered at the orbit path. With each orbit, the satellites observe a new path. SSEC typically receives real-time imagery from the operational POES satellites plus others like Metop, Aqua and Terra.

See the SSEC Satellite Data Service's website for a list of all current, archived (goes) and archived (non-goes) satellite data available from SSEC.

Satellite Data Storage

McIDAS contains various servers that can read and write multiple formats of satellite data. For example, you can setup a dataset to list, display, and manipulate mission-standard Level 1b files in netCDF-4 format or Joint Polar Satellite System (JPSS) Series VIIRS SDR files in HDF5 format. Datasets can also be setup to write GeoTIFF and netCDF files. The help section of the DSSERVE command describes each type of server, and the necessary information to setup the dataset.

Type: **HELP DSSERVE**

When copying satellite data locally to your machine for use in McIDAS, the resulant file format is called *Area*. You can copy, change, display, and delete Area files. Area files contain both data and area directories, similar to how each satellite server assembles the data in memory for use in McIDAS. These satellite data can be displayed in McIDAS image frames. The server or area directory contains descriptive information, such as the sensor source, image date, picture start time, and image coordinates. To see this information, use the IMGLIST command, as shown below to list a mission-standard Level 1b file in netCDF-4 format setup in a dataset which is using the ABIN server format:

```
IMGLIST G16C.1 FORM=ALL Image file directory listing for:G16C Pos Satellite/ Date Time Center Res (km) Image_Size sensor
Lat Lon Lat Lon --- ------------- ------------ -------- ---- ---- ----- ----- ------------ 1 GOES-16 14 SEP 19257
10:51:13 30 87 Band: 1 0.47 um VIS aerosol-over-land 1.00 1.00 3000 x 5000 Band: 2 0.64 um VIS clouds fog, insol, winds
0.50 0.50 6000 x10000 Band: 3 0.86 um Near IR veg/burn scar,aerosol, w 1.00 1.00 3000 x 5000 Band: 4 1.37 um Near IR
cirrus cloud 2.00 2.00 1500 x 2500 Band: 5 1.6 um Near IR cloud phase, snow 1.00 1.00 3000 x 5000 Band: 6 2.2 um Near IR
land/cloud, vege, snow 2.00 2.00 1500 x 2500 Band: 7 3.9 um IR Sfc, cloud, fog, fire, winds 2.00 2.00 1500 x 2500 Band: 8
6.2 um IR High-level WV, winds, rainfall 2.00 2.00 1500 x 2500 Band: 9 6.9 um IR Mid-level WV, winds, rainfall 2.00 2.00
1500 x 2500 Band: 10 7.3 um IR Lower-level WV, winds & SO2 2.00 2.00 1500 x 2500 Band: 11 8.4 um IR Total WV cloud phase,
dust 2.00 2.00 1500 x 2500 Band: 12 9.6 um IR Total ozone,turbulence,wind 2.00 2.00 1500 x 2500 Band: 13 10.3 um IR
Surface & cloud 2.00 2.00 1500 x 2500 Band: 14 11.2 um IR Imagery,SST,clouds,rainfall 2.00 2.00 1500 x 2500 Band: 15 12.3
um IR Total water, ash, and SST 2.00 2.00 1500 x 2500 Band: 16 13.3 um IR Air temp, cloud hgt and amt 2.00 2.00 1500 x
2500 proj: 0 created: 2019257 105113 memo: GOES ConUS - Mode 6 type:ABIN cal type:RAW offsets: data= 1280 navigation= 256
calibration= 768 auxiliary= 0 doc length: 0 cal length: 0 lev length: 0 PREFIX= 0 valcod: 0 zcor: 0 avg-smp: A lcor: 1
ecor: 1 bytes per pixel: 2 ss:186 Resolution Factors (base=1): Line= 4.0 Element= 4.0 IMGLIST: done
```

Area Naming Conventions

Area files use the naming convention *AREAnnnn* where *nnnn* is the four digit area number. For example, AREA0003 is the name of the file that contains area 3. In ADDE, sequences of Area files are grouped together in a dataset. For example, in a previous lesson, you created the dataset MYDATA/IMAGES, which contains the Area file numbers from 1 to 9999. Individual Area files can be accessed with their position number within the dataset. In the case of MYDATA/IMAGES, position 3 (MYDATA/IMAGES.3) would relate to the file AREA0003. However, in the dataset MYDATA/TEST-IMAGES that was created in the previous lesson, the dataset defined the Area files ranging from 4000 to 4004, so position 3 (MYDATA/TEST-IMAGES.3) would be AREA4002. (This would be equivalent to MYDATA/IMAGES.4002)

Coordinate Systems

McIDAS references image data in four different, but interconnected coordinate systems:

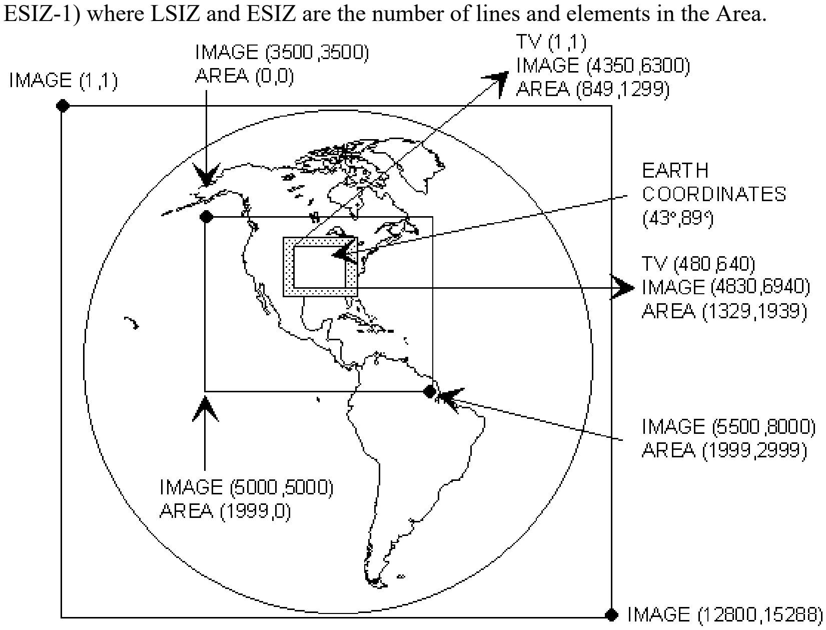
- Image
- Area (File)
- Earth
- TV

Image Coordinates

The image coordinate system forms the basis for the other McIDAS coordinate systems. A full image is a sequence of lines and elements arranged from top to bottom. The top line and leftmost element have image coordinates (1,1). Therefore, each pixel has a unique pair of line and element values that are its image coordinates. The figure below represents a full image, image sector and displayed Area. The upper-left image coordinates of the full image are (1,1) and the upper-left coordinates of the image sector are (3500,5000).

Area (File) Coordinates

Area coordinates (file coordinates in ADDE) are based on the size of the Area only. Like image coordinates, area coordinates are referenced as lines and elements. The first pixel has area coordinates (0,0) as shown in the image sector below. The bottom-right pixel has area coordinates (LSIZ-1, ESIZ-1) where LSIZ and ESIZ are the number of lines and elements in the Area.



Earth Coordinates

If the displayed image is navigated, the image coordinates can be converted to earth coordinates (latitude and longitude). Earth coordinates are specified in degrees, minutes, and seconds in the form DDD:MM:SS. Southern latitudes and longitude **east** of Greenwich are negative. Latitudes run from -90:00:00 to +90:00:00 and longitudes run from -180:00:00 to +180:00:00.

TV Coordinates

The pixels on the McIDAS image frames are arranged by raster lines and pictel elements. The raster lines run horizontally across the frame and the pictel elements run vertically across the frame. The pixel in the upper-left corner of the frame is numbered (1,1) which means (raster line 1, pictel element 1). The total number of raster lines and pictel elements on the frame is determined by the frame size. The lower-right corner of the default-sized frame is (480,640) in TV coordinates.

Image Navigation

Navigation, as applied to satellite imagery, means the conversion between satellite image coordinates (line and element) and earth coordinates (latitude and longitude). This is usually done when the data is ingested.

If a displayed image sector is navigated, McIDAS can convert the image coordinates of a specified pixel to earth coordinates. The E command lists the earth, TV and image coordinates at the cursor center.

Image Resolution

Image resolution is measured in kilometers and is highest at the subsatellite point. The highest available resolution depends on the satellite, band and image type:

Satellite	Visible	Infrared
GOES	0.5 km	2 km
HIMAWARI	0.5 km	2 km
MODIS	0.25 km	1 km
MSG	1 km	3 km
POES	1 km	1 km

This means that a single GOES infrared pixel at the subsatellite point represents a 2km x 2km square on the earth's surface, and a single MSG visible pixel at the subsatellite point represents a 1km x 1km square on the earth's surface, etc. The further a pixel is located from the subsatellite point, the lower its resolution.

You can display imagery at a resolution other than that stored in the Area. Specifying a positive magnification factor in McIDAS commands (e.g., IMGDISP) enlarges or *blows up* the data by replicating pixel values; a negative magnification factor produces a *blow down* by sampling pixel values. For example, if you choose a magnification factor of 16, the value of each pixel in the Area is duplicated 256 times (in a 16 x 16 box of pixels) when displayed on the frame. If you choose a magnification factor of -4, the value of every fourth element along every fourth line is displayed as one pixel on the image frame.

Other Digital Imagery

Although geostationary satellites, polar orbiting satellites and radar images are usually displayed, you can digitize any data and display it on McIDAS. For example, cell proteins have been digitized for biochemistry studies and human

Satellite Imagery - Displaying Satellite Data

Before displaying satellite data, you must know the ADDE dataset name under which it is stored. The DSINFO and IMGLIST commands can list information about image datasets you have access to. Once located, you can display the images using the IMGDISP command.

When you display an image on a frame, McIDAS creates a frame directory for the image which lists the frame number, sensor source, date, image time, band number, image coordinates, magnification factors, and ADDE dataset name and file position number. The FRMLIST command lists the frame directory.

You can display images one at a time or in a sequence. If the areas on disk are chronologically ordered, an entire set of images can be sequentially loaded using one IMGDISP command with a repeat factor.

In this exercise, you will select two images and display them on frames, list the frame directory for one frame, and display a sequence of images using the IMGDISP command with a repeat factor.

1. Start a McIDAS session.
   At the Unix prompt:
   Type: **`mcidas`**
   Your session should still be set for twenty frames from the last time you changed the Configuration GUI. If not, set it for twenty frames, as described in Getting Started.
2. Setup a new dataset of 10 Mesoscale ABI images of Hurricane Dorian. Download a set of Level 1B netCDF files from a FTP site. For this example, it is assumed the data has been placed in the directory ***<local-path>/Data/Satellite/ABI_data/Meso1.***
   Download the files from ftp.ssec.wisc.edu.
   ftp ftp.ssec.wisc.edu
   login anonymous
   prompt
   binary
   cd pub/mug/Data/Satellite/ABI_data/Meso1
   mget OR_ABI*
   Example file name:
   OR_ABI-L1b-RadM1-M6C02_G16_s20192461800251_e20192461800309_c20192461800346.nc
3. Run the DSSERVE command to access the Level 1b netCDF imagery files.
   Type: **DSSERVE ADD DORIAN/M1 ABIN TYPE=IMAGE DIRFILE='<local-path>/Data/Satellite/ABI_data/Meso1 /OR_ABI*'**
4. List the area directories for the first 3 images in the DORIAN/M1 dataset.
   Type: **IMGLIST DORIAN/M1.1 3**

```
Image file directory listing for:DORIAN/M1 Pos Satellite/ Date Time Center Band(s) sensor Lat Lon --- -------------
------------ -------- ---- ---- ------------ 1 GOES-16 3 SEP 19246 18:00:25 28 79 2 2 GOES-16 3 SEP 19246 18:01:22
28 79 2 3 GOES-16 3 SEP 19246 18:02:22 28 79 2
```

5. List the directories of all images of the DORIAN/M1 dataset that contain images between 18:01:00 and 18:05:00 UTC on day 2019246.
   Type: **IMGLIST DORIAN/M1.ALL DAY=19246 TIME=18:01:00 18:05:00**

```
IMGLIST DORIAN/M1.ALL DAY=19246 TIME=18:01:00 18:05:00 Image file directory listing for:DORIAN/M1 Pos Satellite/
Date Time Center Band(s) sensor Lat Lon --- ------------- ------------ -------- ---- ---- ------------ 2 GOES-16 3
SEP 19246 18:01:22 28 79 2 3 GOES-16 3 SEP 19246 18:02:22 28 79 2 4 GOES-16 3 SEP 19246 18:03:22 28 79 2 5 GOES-16 3
SEP 19246 18:04:22 28 79 2
```

6. Display the first GOES-16 visible 0.5km image on frame 1.
   Type: **IMGDISP DORIAN/M1.1 1**
   Notice that the bottom of the image has an annotation line which lists the frame number, satellite type, sensor source, Gregorian date, Julian date, UTC time, upper-left corner image line and element, and resolution. Note the Note the resolution of 1.0, this is the base resolution (best) of the satellite, so it lists 1.0, not 0.5 in the frame label. Run the following command to confirm this:
   Type: **IMGLIST DORIAN/M1.1 1 FORM=ALL**
7. List the frame directory for the current image on frame 1.
   Type: **FRMLIST**
   Information about the frame number, sensor source, band, date, time, and ADDE dataset/position number is displayed for the image.

```
FRMLIST Frame Satellite [panel] Sensor Band Date Time Data Source --------- ------------ ---- ------------ --------
----------------------------- 1[1] GOES-16 2 03 Sep 19246 18:00:25 DORIAN/M1.1 FRMLIST: Done
```

8. Display three areas that are in sequence using one IMGDISP command.
   Type: **IMGDISP DORIAN/M1.1 2 REPEAT=3;SF 2**
   The repeat factor is 3; therefore this entry displays three images in sequence, on frames 2, 3, and 4.

9. Step through the frames to view the images.
   Press: **Alt A**
   Press: **Alt B**
10. Erase the contents of frames 2 through 4 to prepare for the next exercise.
    Type: **ERASE I 2 4**

Satellite Imagery - Using Coordinate Systems

In this exercise, you will display the same image using different coordinate types. The coordinate system and location where the specified coordinates are positioned on the frame define the coordinate type. The three coordinate systems are:

- Earth
- File (Area)
- Image

The two locations (PLACE) are:

- CENTER (center)
- ULEFT (upper-left corner)

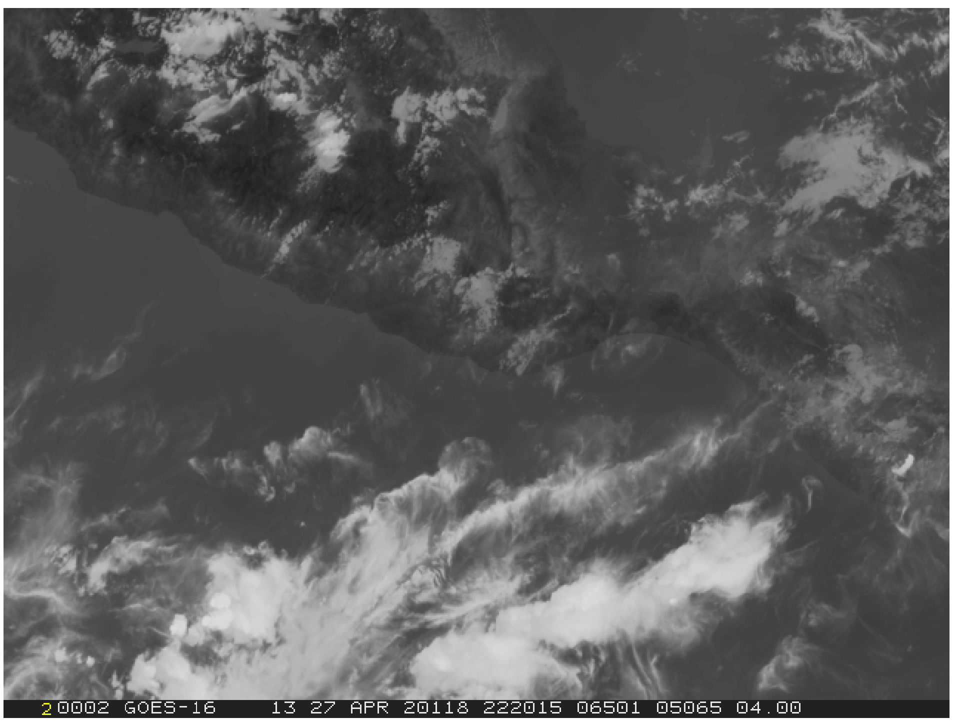
The coordinate type is defined by a combination of the coordinate system keywords and location; for example, LINELE=6261 4745 I PLACE=ULEFT positions the (I) image coordinates (6261,4745) in the (ULEFT) upper-left corner.

The first exercise shows that the same image can be displayed using different coordinate types.

1. Show and erase frame 1.
   Type: **SF 1;ERASE X 1**
2. Display the GOES-16 Full Disk Infrared image on frame 1 so the pixel having image coordinates (6501,5065) appears at the upper-left corner of the frame.
   Type: **IMGDISP G16FD.-1 1 LINELE=6501 5065 I PLACE=ULEFT BAND=13**
   The combination of I for line/element type and PLACE=ULEFT places the specified image coordinates in the upper-left corner. Note in the on-line help for IMGDISP that the default for **PLACE** is ULEFT when **LINELE** is specified and therefore could be omitted from the command line above.

3. Display the GOES-16 Full Disk Infrared image on frame 2 so that Puerto Escondido, Mexico, is at the center of the image frame.
   Type: **IMGDISP G16FD.-1 2 STATION=MMPS BAND=13;SF 2**

   The STATION parameter places the earth coordinates for Puerto Escondido in the center of the frame, as shown below. When STATION or LATLON are used, the default for **PLACE** is CENTER.

4. Display the GOES-16 Full Disk Infrared image on frame 3 so that file (area) coordinates (1865, 1586) appear in the center of the image frame.
   Type: **IMGDISP G16FD.-1 3 LINELE=1865 1586 F PLACE=CENTER BAND=13;SF 3**
   The combination of F for coordinate system and PLACE=CENTER places the specified file (area) coordinates in the center of the image frame.

5. Start and then stop the loop.
   Press: **LS 1-3;LS O 4-6**
   Press: **Alt L**

Press: **Alt L**

All the images are the same, except they are displayed using different coordinate types.

6. Show frame 2 and use the PC command to position the cursor in the center the frame. Then find the earth coordinates of Puerto Escondido. The PC command is discussed in more detail in the Graphics and the Cursor lesson.
   Type: **SF 2;PC C**
   Press: **Alt E**
   Command E lists the latitude and longitude of a pixel on a navigated frame in the format DD:MM:SS. It also lists the TV coordinates and image coordinates of the pixel.
   ```
   Frame Latitude Longitude Tvline Tvelem Line Elem 2 15:52:12 097:05:00 240 320 7457 6341
   ```

Next you will display the same image three times, but change the location of the displayed image using different coordinate types.

1. Erase frames 1 through 3 and show frame 1.
   Type: **ERASE I 1 3;SF 1**
2. Display the visible image from the ABI/CONUS dataset on frame 1 so that the earth coordinates 30° latitude and 89° longitude are centered in the frame. Then position the cursor at the center of the frame and verify that the earth coordinates are 30° latitude and 89° longitude.
   Type: **IMGDISP ABI/CONUS 1 LATLON=30 89 BAND=2**
   Type: **PC C**
   Press: **Alt E**
   Notice that the center of the frame has a latitude of 30:00:07 and a longitude of 89:00:07.

3. Display the image on frame 2 so that the image coordinates (1999, 2503) are centered in the frame and show frame 2. Then position the cursor in the center of the frame.
   Type: **IMGDISP ABI/CONUS 2 BAND=2 LINELE=1999 2503 I PLACE=CENTER;SF 2**
   Type: **PC C**
   Press: **Alt E**
   Notice that the image coordinates (1999, 2503) are in the center of the frame.

4. Display the image on frame 3 so that the file (area) coordinates (0,0) appear in the upper-left corner of the frame and show frame 3.
   Type: **IMGDISP ABI/CONUS 3 BAND=2 LINELE=0 0 MAG=-4;SF 3**
5. Change the dwell rate and then start and stop the loop. Notice how the image position changes.
   Type: **DR 10**
   Press: **Alt L**
   Press: **Alt L**

Next, you will display a sequence of real time GOES-16 mid-level water vapor and infrared images using the same earth coordinates to compare them. Display the GOES water vapor images on the primary loop and the GOES infrared images on the opposite loop.

1. Display the first GOES-16 0.5km water vapor image on frame 1, centering the image on earth coordinates 30° N and 75° W. Use the SF= keyword to automatically display the frame.
   Type: **IMGDISP G16FD.1 1 LATLON=30 75 BAND=9 SF=YES**
2. Display the next two GOES-16 water vapor images on frames 2 and 3, centering the images on coordinates 30° N and 75° W.
   Type: **IMGDISP G16FD.2 2 LATLON=30 75 BAND=9; IMGDISP G16FD.3 3 LATLON=30 75 BAND=9**
3. Display the three GOES-16 infrared images on frames opposite the GOES-16 visible images. The repeat factor (3) loads all three images with one IMGDISP command.
   Type: **SF 1;IMGDISP G16FD.1 OPP LATLON=30 75 BAND=13 REPEAT=3**
   You are using the same coordinates, 30° N and 75° W, to load the images. The repeat factor 3 displays three images in sequence, on frames 4, 5, and 6.

4. Change the dwell rate and loop the images.
   Type: **DR 5**
   Press: **Alt L**
   You should see the loop of the GOES water vapor images on frames 1 through 3.

5. Toggle to the opposite loop to compare the images. Then stop the loop.
   Press: **Alt O**
   Press: **Alt L**
6. Practice comparing the images with the loop control system commands Alt A, Alt B, Alt L, and Alt O that you learned in the Loop Control System lesson.

Satellite Imagery - Listing Image Data

In this exercise, you will list image data by positioning the cursor on the image and entering the IMGPROBE command.

The IMGPROBE command lists data from the area represented on the image at the cursor center. It lists the area number, area coordinates, image coordinates, raw, brightness, and when present, temperature and radiance values. IMGPROBE can also be invoked by pressing Alt-D.

When the *region* parameter is specified as **BOX**, the IMGPROBE command lists the data from the area inside the cursor. By specifying different parameters, you can list raw, brightness, and when present, temperature and radiance values.

Note: If a new image becomes available on the server, the IMGPROBE commands in this exercise may error. The image will need to be reloaded with the IMGDISP command.

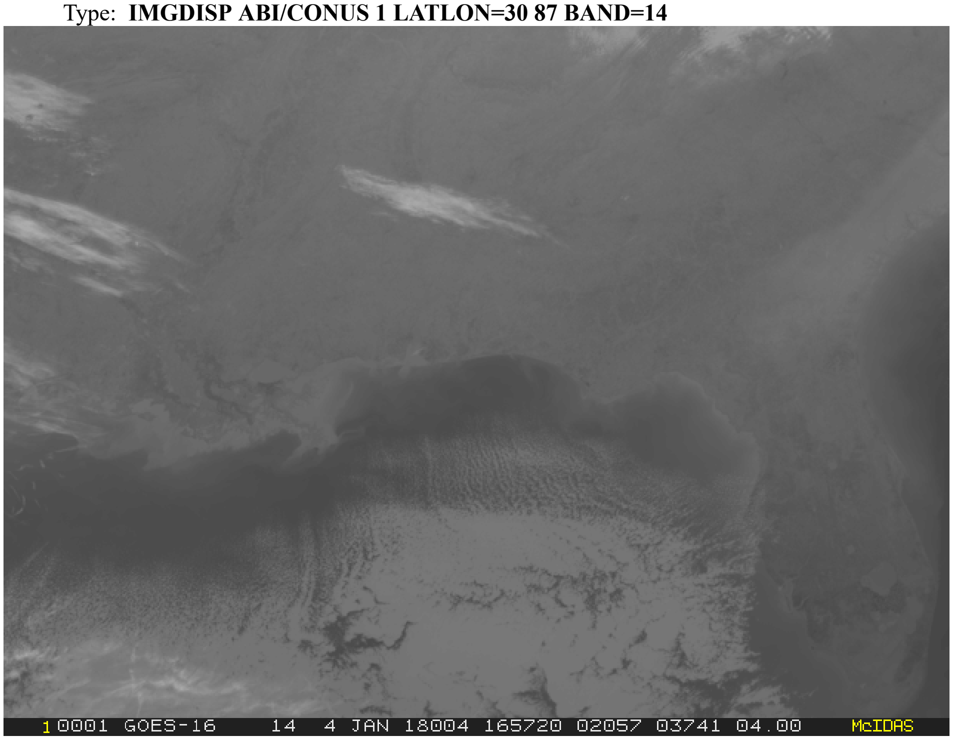
1. Change your cursor to a smaller size with the CUR command, which is explained in more detail in the Graphics and the Cursor lesson.
   Type: **CUR 5 5**
2. Show the GOES-16 infrared image on frame 6 and position the cursor in the center of the frame. List the area values for the element at the cursor center by using the IMGPROBE command in the non-interactive mode.
   Type: **SF 6;PC C;IMGPROBE MODE=N**
   Alternate method:
   Type: **SF 6;PC C**
   Press: **Alt D**
   The area number, area coordinates, image coordinates, raw, and brightness values are listed. The Nominal Time is the time the scan began, and the Scan Time is the time that pixel was scanned. Because this is an infrared image, the temperature and brightness values are also listed.
   ```
   Image Name Day Nominal Time Scan Time Band ---------------- ------- ------------ --------- ---- G16FD.12 28 Apr
   20119 11:00:15 11:01:50 13 File Nominal Image RAW RAD TEMP BRIT Lat/Lon Line/Element Line/Element * K 30:00:20/
   74:59:31 1170/ 2712 4681/10849 2065 92.786 292.23 76 * milliwatts/meter**2/steradian/(cm-1)
   ```
3. List the brightness values for the area inside the cursor.
   Type: **PC C;IMGPROBE LIST BOX BRIT MODE=N**

   Notice that the data value at the cursor center is the same value listed as with IMGPROBE.
4. Display a GOES-16 visible image on frame 1 and list the values at the center of the cursor.
   Type: **SF 1**
   Type: **IMGDISP G16FD.-1 LAT=30 75 BAND=2**
   Press: **Alt D**
   For GOES-16 visible data, albedo values are listed with the radiance, raw and brightness; whereas, temperature, radiance, raw and brightness values were listed for the infrared image in step 3.

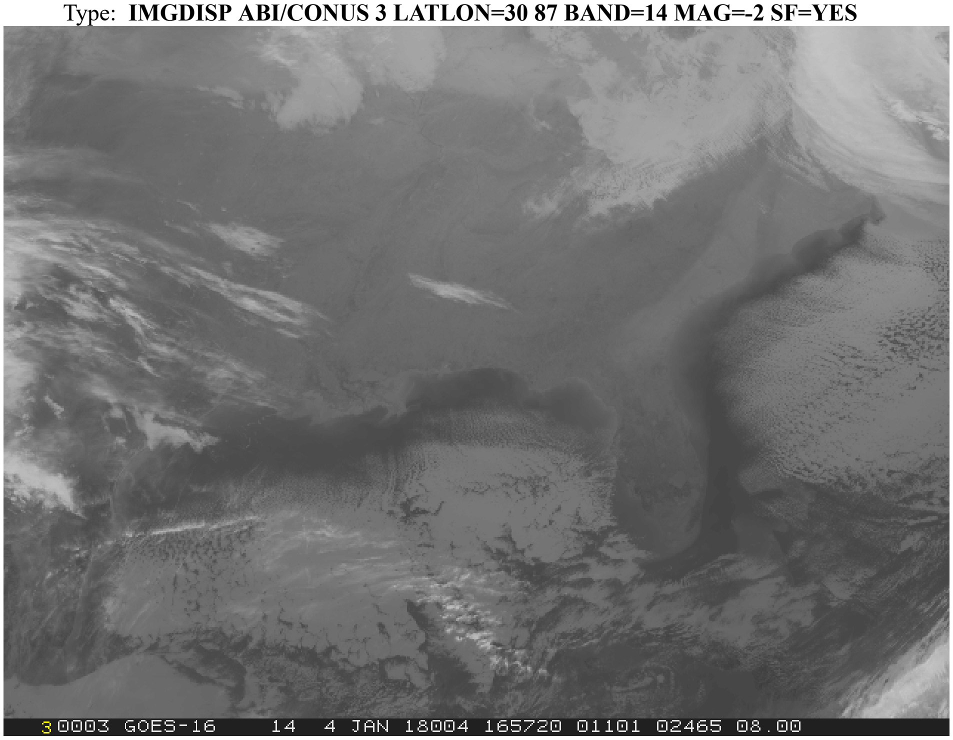
Satellite Imagery - Changing the Image Resolution

In this exercise, you will magnify the resolution of displayed images using the IMGDISP command. Then, you will use the IMGPROBE command to list data values inside the cursor. Many other commands, for example IMGCOPY, have parameters to change the resolution.

1. Display the local GOES-16 conus image, band 14, in its original resolution (2 km) on frame 1. Center the image on earth coordinates 30° and 87°
   Type: **IMGDISP ABI/CONUS 1 LATLON=30 87 BAND=14**



2. Decrease (blow down) the image resolution of the GOES-16 conus image by a factor of 2 and display it on frame 3.
   Type: **IMGDISP ABI/CONUS 3 LATLON=30 87 BAND=14 MAG=-2 SF=YES**



3. Magnify (blow up) the image resolution of the GOES-16 conus image by factor of 2 and display it on frame 2.
   Type: **IMGDISP ABI/CONUS 2 LATLON=30 87 BAND=14 MAG=2 SF=YES**

4. Loop the frames.
   Press: **Alt L**
5. Change the loop sequence to view the images in the order of increasing resolution.
   Type: **LS 3 1 2**
6. Stop the loop.
   Press: **Alt L**
7. Show frame 1 and list the data and brightness values inside the cursor.
   Type: **PC C;SF 1;IMGPROBE LIST BOX BRIT MODE=N**

   Notice you have 5 lines and 5 elements of data.

8. Show frame 3 and list the data and brightness values inside the cursor.
   Type: **SF 3;IMGPROBE LIST BOX BRIT MODE=N**

   Notice that you have 10 lines and 10 elements of data. Because the image was blown down by a factor of 2, every second line and element was sampled. Therefore, the number of lines and elements read from the area is twice the amount displayed as an image. The following equation determines the number of lines and elements in the cursor area for a blow down:

   *cursor size***blow down factor*

   which for this example is 5 * 2 = 10.

9. Show frame 2 and list the data and brightness values inside the cursor.
   Type: **SF 2;IMGPROBE LIST BOX BRIT MODE=N**

   There are 3 lines by 3 elements of data listed. The following equation determines the number of lines and elements in the cursor area for a blow up:

   (*cursor size*-1)/*blow up factor* + 1

   which for this example is [(5 - 1)/2] + 1 = 3.

Satellite Imagery - Copying and Displaying Images

In this exercise, you will copy and display areas with the IMGCOPY and IMGDISP commands.

1. Show frame 4, copy a GOES-16 0.5km Vis image to position 1 in your MYDATA/TEST-IMAGES (AKA TI) dataset, and display the image on frame 4. The SIZE=SAME keyword copies the entire image to the destination dataset.
   Type: **SF 4**
   Type: **IMGCOPY ABI/CONUS TI.1 BAND=2 SIZE=SAME; IMGDISP TI.1 4 LAT=30 87**
2. List the two image directories
   Type: **IMGLIST ABI/CONUS FORM=BAND;IMGLIST TI.1 FORM=BAND**
   Notice that the images now contain the same information and the band 2 image resolutions are nearly the same.
   ```
   Image file directory listing for:ABI/CONUS Pos Satellite/ Date Time Center Res (km) Image_Size sensor Lat Lon Lat Lon --- ------------- ------------ -------- ---- ---- ----- ----- ------------ 1 GOES-16 4 JAN 18004 16:57:20 30 87 Band: 2 0.64 um VIS clouds fog, insol, winds 0.64 0.54 6000 x10000 Band: 14 11.2 um IR Imagery,SST,clouds,rainfall 2.55 2.13 1500 x 2500 Image file directory listing for:TI Pos Satellite/ Date Time Center Res (km) Image_Size sensor Lat Lon Lat Lon --- ------------- ------------ -------- ---- ---- ----- ----- ------------ 1 GOES-16 4 JAN 18004 16:57:20 30 87 Band: 2 0.64 um VIS clouds fog, insol, winds 0.63 0.54 6000 x10000
   ```
   The Latitude and Longitude values listed above are the actual resolution values at the center point of the image.

3. Show frame 5, copy the first GOES-16 2km IR image to the second position in your TI dataset, and display the image on frame 5 centered on Boston.
   Type: **SF 5**
   Type: **IMGCOPY ABI/CONUS.1 TI.2 BAND=14 STA=KBOS; IMGDISP TI.2 5**
4. List the two image directories.
   Type: **IMGLIST ABI/CONUS.1 FORM=BAND;IMGLIST TI.2 FORM=BAND**
   In ABI/CONUS.1, the image size of band 14 is 1500 x 2500. Notice that the image size for TI.2 is 480 x 640. If the SIZE= keyword is not specified, a 480 by 640 image sector (the default size of image frames) is copied to the new area.
   ```
   Image file directory listing for:ABI/CONUS Pos Satellite/ Date Time Center Res (km) Image_Size sensor Lat Lon Lat Lon --- ------------- ------------ -------- ---- ---- ----- ----- ------------ 1 GOES-16 4 JAN 18004 16:57:20 30 87 Band: 2 0.64 um VIS clouds fog, insol, winds 0.64 0.54 6000 x10000 Band: 14 11.2 um IR Imagery,SST,clouds,rainfall 2.55 2.13 1500 x 2500 Image file directory listing for:TI Pos Satellite/ Date Time Center Res (km) Image_Size sensor Lat Lon Lat Lon --- ------------- ------------ -------- ---- ---- ----- ----- ------------ 2 GOES-16 4 JAN 18004 16:57:20 42 71 Band: 14 11.2 um IR Imagery,SST,clouds,rainfall 3.21 2.12 480 x 640
   ```
5. Show frame 6, copy a GOES-16 2km Infrared image to TI.3 centered on Mobile, Alabama, and increase the resolution by a factor of 2. Display it on frame 6.
   Type: **SF 6;IMGCOPY ABI/CONUS.1 TI.3 STATION=MOB BAND=14 MAG=2;IMGDISP TI.3 6**
   The image is displayed on frame 6 with a resolution of 1 km (2km is listed due to the base resolution of the satellite being 0.5km).

6. List the two images to compare image resolutions. Note that the lat and lon resolution values are different between the images.
   Type: **IMGLIST ABI/CONUS.1 BAND=14 FORM=BAND;IMGLIST TI.3 FORM=BAND**
   The original image (ABI/CONUS.1 BAND=14) had a resolution of ~2 km and the new area (TI.3) has a resolution of ~1 km. This is a blow up.

7. Display a loop of GOES-16 Meso1 images from 17 to 18 UTC on frames 4 through 6 and use the REFRESH keyword to plot a high resolution map on each of the images.
   Type: **IMGDISP RTGOESR/M1C10 TIME=17 18 ALL=4 6 REFRESH='MAP VH GRA=(GRA) IMA=(IMA)'**
   Notice that the numbers placed into the MAP command's GRA and IMA keywords take the place of (GRA) and (IMA) and now match the image frame numbers of the loop.

Satellite Imagery - Manipulating Images

A major strength of McIDAS as a meteorological data analysis/display package is its ability to display and manipulate satellite images. In this lesson, you will use the IMGREMAP and IMGOPER commands to remap images and create new image products.

1. Remap the second latest GOES-16 Visible image in the RTGOESR/FD dataset to a mercator projection centered on Washington DC, and place it in the fourth position in the TI dataset. Set the output resolution at 4 km. IMGREMAP calculates the size needed to make the default 480x640 image.
   Type: **IMGREMAP RTGOESR/FD.-1 TI.4 STATION=DCA BAND=2 PRO=MERC RES=4**
2. List out the directory information for these images.
   Type: **IMGLIST RTGOESR/FD.-1 BAND=2 FORM=BAND;IMGLIST TI.4 FORM=BAND**

```
Image file directory listing for:RTGOESR/FD Pos Satellite/ Date Time Center Res (km) Image_Size sensor Lat Lon Lat
Lon --- ------------- ------------ -------- ---- ---- ----- ----- ------------ 192 GOES-16 17 SEP 19260 10:00:18 0
75 Band: 2 0.64 um VIS clouds fog, insol, winds 0.50 0.50 21696 x21696 191 GOES-16 17 SEP 19260 09:50:18 0 75 Band:
2 0.64 um VIS clouds fog, insol, winds 0.50 0.50 21696 x21696 Image file directory listing for:TI Pos Satellite/
Date Time Center Res (km) Image_Size sensor Lat Lon Lat Lon --- ------------- ------------ -------- ---- ---- -----
----- ------------ 4 GOES-16 17 SEP 19260 09:50:18 39 77 Band: 2 0.64 um VIS clouds fog, insol, winds 3.12 3.12 480
x 640
```

   Notice that the remapped image now has a different center lat/lon and is a different size than the original image.

3. Show frame 1. Display the images in frames 1 and 2 centered over Washington DC. Draw a map of dashed lat/lon lines in color 3 on each image. You will learn more about the MAP command in the Graphics and the Cursor lesson.
   Type: **SF 1**
   Type: **IMGDISP RTGOESR/FD.-1 1 BAND=2 STATION=DCA REFRESH='MAP X X LALO -3 GRA=(GRA) IMA=(GRA)'**
   Type: **IMGDISP TI.4 2 STATION=DCA REFRESH='MAP LALO -3 GRA=(GRA) IMA=(GRA)'**
4. Set the loop sequence, and loop between the frames.
   Type: **LS 1-2**
   Press: **Alt L**
   Notice that the images have different projections.

5. Stop the loop.
   Press: **Alt L**
6. Create a new image using two DORIAN visible GOES-16 images in the DORIAN dataset using the formula DORIAN/M1.1 - DORIAN /M1.10. This will show the shift in the clouds from one image to the next. Put the new image into position 5 of the TI dataset.
   Type: **IMGOPER DORIAN/M1.1 DORIAN/M1.10 TI.5 COEF=1 -1 FORM=ADD MAG=-2**

   IMGOPER generates a new image by applying mathematical functions to data from one or more source images. The following equation computes the data value for each line/element pair in the destination image. This operation is performed repeatedly using data from source image line/element pairs as input values (*input*) until the entire image is completed.

   output data value=

   FUNC[ACON+(MCON*(FORM((COEF1*(OFF1+((SIGN1)**input1*)**POW1))

   (COEF2*(OFF2+((SIGN2)**input2*)**POW2)) . .

   (COEF*n**(OFF*n*+((SIGN*n*)**inputn*)**POW*n*)))))]

   The part of the equation operating on a single input data value, (COEF*n**(OFF*n*+((SIGN*n*)**inputn*)**POW*n*)) , is referred to as a *term. n* represents the number of *sdataset* images. It may not be larger than 100. *input1 . . n* represents the individual data values from each of the source images. The **FORM** keyword determines how the terms are combined. For example, if you specify **FORM=MULT**, the terms are multiplied.

7. Erase the maps in graphics frames 4 through 6, and display the images that you just used in frames 3-5.
   Type: **ERASE G 4 6**
   Type: **IMGDISP DORIAN/M1.1 3 MAG=-2**
   Type: **IMGDISP DORIAN/M1.10 4 MAG=-2**
   Type: **IMGDISP TI.5 5**
8. Loop through the images.
   Type: **LS 3-5**
   Press: **Alt L**
9. Stop the loop.
   Press: **Alt L**
10. Add a label to the image frame.
    Type: **SF 5;FRMLABEL IMA=5 "CLOUD MOVEMENT FROM 18:00:25 TO 18:09:22**

Satellite Imagery - Deleting Images

In this exercise, you will delete several images in a dataset using the IMGDEL command.

1. List the directories of the images created in this lesson, then delete the images.
   Type: **IMGLIST TI.1 5**
   Type: **IMGDEL TI.1 5**
2. List the image directories again.
   Type: **IMGLIST TI.1 5**
   No images are listed for the dataset range given. Once you delete an area, you can no longer access data values such as temperature, even if the image is still displayed. The image is only a representation of the data.

Satellite Imagery - Using Briefing Windows

In addition to displaying loops in the Image Window, you can also create briefing windows to view saved loops. Each briefing window can display a frame or loop different from those currently shown in the Image Window or other briefing windows.

In this exercise, you will display a sequence of GOES-16 visible and infrared image with the same geographical coverage. You will then create two briefing frames, so that you can view the images at the same time.

1. Erase the graphics in frames 1 through 6.
   Type: **ERASE G 1 6**
2. Display the first GOES-16 0.5km Visible image area on frame 1, centering the image on earth coordinates 30° N and 81° W. Use the SF= keyword to automatically display the frame. Note: The data in positions 50 through 52 on the real time server may be night time images for the visible data.
   Type: **IMGDISP G16FD.50 1 LATLON=30 81 BAND=2 MAG=-4 SF=YES**
3. Display the next two GOES-16 visible images on frames 2 and 3, centering the images on coordinates 30° N and 81° W.
   Type: **IMGDISP G16FD.51 2 LATLON=30 81 BAND=2 MAG=-4; IMGDISP G16FD.52 3 LATLON=30 81 BAND=2 MAG=-4**
4. Display the three GOES- 16 infrared images, which begin in position 50, on frames opposite the GOES-16 visible images. The repeat factor (3) loads all three images with one IMGDISP command.
   Type: **IMGDISP G16FD.50 OPP LATLON=30 81 BAND=13 REPEAT=3**
   You are using the same coordinates, 30° N and 81° W, to load the images. MAG=-4 used with the visible display will blowdown band 2 to the same domain as the band 13 images. The repeat factor 3 displays three images in sequence, on frames 11, 12, and 13. Note: When using opposite frames on a session other than 20 frames, the opposite frame numbers may vary.

5. Set the loop, change the dwell rate, and loop the images.
   Type: **LS 1-3**
   Type: **DR 5**
   Press: **Alt L**
   You should see the loop of the GOES-16 visible images on frames 1 through 3.

6. Toggle to the opposite loop to compare the images. Then stop the loop.
   Press: **Alt O**

   You should see the loop of the GOES-16 infrared images on frames 11 through 13.

7. Create a text file that will be used to create briefing frames.
   Save the file BRIEF.SYNC in your $HOME/mcidas/data directory as a text file.
8. Create two briefing windows that will have frames 1-3 in one briefing window, and frames 11-13 in the other briefing window.
   Type: **SYNC BRIEF.SYNC**
9. Close the briefing windows by clicking on the upper-left corner menu bar, and selecting CLOSE from the window menu.

10. Exit McIDAS
    Type: **EXIT**

Graphics and the Cursor

Table of Contents:

Graphics and the Cursor

Graphics are the line drawings such as maps, isolines, diagrams, or text displayed on an image frame. You can change the graphics color, line width, dash length, dash gap length, and dash gap color.

The cursor is a visible marker on an image frame that can be moved with the mouse. You can use the cursor to point to specific locations and then determine the coordinates. You can change the color, shape, and size of the cursor.

This lesson describes how to:

- define graphics parameters
- generate graphics
- erase graphics
- save and restore graphics and frames
- change the cursor shape, color and size
- contour data with the cursor
- zoom in on frames with the cursor
- locate specific positions using the cursor
- find distances between points

The following commands are used in this lesson.

Command	Function
CCODE	lists country codes
CUR	defines the cursor size, shape and color
CW	fills or erases graphics inside the cursor
DIST	finds the distance on a navigated frame
DMAP	lists information about files
E	lists earth, image and TV coordinates
ERASE	erases graphics, images or entire frames
FRMSAVE	saves a McIDAS frame to a GIF, JPEG, PostScript, color PostScript, PPM, or BMP format file
GD	defines graphics display parameters
GU	graphics utility
IMGPROBE	lists or plots image data for a region of the displayed image
K	toggles the image frame on and off
LVF	lists virtual frame files

MAP	draws maps and latitude/longitude lines
PC	positions the cursor at a specific location
RVF	restores virtual frames saved with command SVF
S	lists Metar, Synoptic and RAOB stations within the cursor bounds
SHOWVG	displays virtual graphics
STNLIST	lists weather stations
STNPLOT	plots stations on a navigated frame
SVF	saves frames to a virtual frame file
TOWNS	lists the U.S. towns closest to the cursor location
W	toggles the graphics frame on and off
Z	toggles the zoom function on and off
ZA	displays an annotation at the cursor
ZLM	draws graphics using the mouse
ZM	sets the zoom factor to use with the Z command

Graphics and the Cursor - Basic Concepts

Graphics

Graphics are lines and text that can be drawn on frames. Examples of graphics include:

- a map of coastal borders overlaying a satellite image
- a plot of surface temperatures over the eastern United States
- a label to identify a satellite image
- a contour of satellite brightness data

A frame can have several different graphics. For example, you can have a map and a temperature contour over a satellite image. By default, McIDAS graphics in McIDAS are independent from the displayed image in a frame. You can remove the graphics in a frame without erasing the underlying image.

Graphics Parameters

You can modify graphics colors using the GU command. Line width, dash and gap lengths, and the color of gaps in dashed lines can also be changed with the GD command.

Multiple colors are particularly useful for creating graphics displays with several overlays. The graphics commands assign colors through the use of *levels*. Each level is assigned one color. For example, by default, level 1 is magenta. The assigned colors are not fixed and you can assign different colors to the levels.

When the color of a level changes, all the graphics already drawn in that level are changed on the current frame.

By default, McIDAS-X has 35 predefined colors and 16 graphics color levels. Therefore, only the maximum number of color levels can be used at one time. However, you can change the color assigned to each level.

The following tables show the default color levels:

```
LEVEL COLOR BLUE GREEN RED ----- ---------- ---- ----- --- 0 BLACK 0 0 0 (Graphic background) 1 MAGENTA 255 0 255 2 CYAN 255 255 0 3 YELLOW 0 255 255 4 GREEN 0 255 0 5 RED 0 0 255 6 BLUE 255 0 0 7 WHITE 255 255 255 8 GRAY 127 127 127 9 GOLD 0 187 255 10 PINK 127 127 255 11 AQUAMARINE 147 219 112 12 ORCHID 219 112 219 13 NAVY 115 0 0 14 SKY 255 163 0 15 BEIGE 127 171 255 16 PURPLE 127 0 127
```

Most McIDAS graphics commands are displayed with a default color level, but you can usually specify a different color level using a parameter or keyword.

Although McIDAS has predefined graphics colors, you can create custom colors using the GU command and specify intensities for each of the primary colors. The intensities range from 0 to 255; 0 is the darkest shade of the color and 255 is the brightest. The predefined colors and their intensities of blue, green and red are listed below. You can save new colors to a file called a graphics table which you can restore to an individual frame or a range of frames.

McIDAS has the following predefined colors:

Color	Blue	Green	Red
Aquamarine	147	219	112
Avocado	35	131	67
Beige	127	171	255
Black	0	0	0
Blue	255	0	0
Brown	35	35	127
Coral	80	127	255
Cyan	255	255	0
Firebrick	35	35	142
Gold	0	187	255
Goldenrod	112	219	219
Gray	127	127	127

Color	Blue	Green	Red
Grey	127	127	127
Green	0	255	0
Khaki	95	159	159
Lemon	67	255	227
Magenta	255	0	255
Maroon	105	35	142
Navy	115	0	0
Orange	0	127	255
Orchid	219	112	219
Pink	127	127	255
Plum	234	173	234
Purple	127	0	127

Color	Blue	Green	Red
Red	255	0	0
Salmon	66	66	111
Sienna	35	107	142
Sky	255	0	0
Tan	112	147	219
Thistle	216	191	216
Turquoise	234	234	173
Violet	203	127	255
Yellow	0	255	255
Wheat	191	216	216
White	255	255	255

Drawing Maps

You can draw maps on frames containing nothing, an image and/or a graphic. To draw a map over a satellite image, the satellite image must be navigated; all real-time satellite images are navigated.

Virtual Graphics and Virtual Frame files

Graphics can be saved in eight formats:

- virtual graphics files
- virtual frame files
- bitmap (BMP)
- color PostScript (CPS)
- graphics interchange format (GIF)
- joint photographic experts group (JPG)
- portable pixmap (PPM)
- PostScript (PS)

Virtual graphics files contain only graphics. Virtual graphics are saved when they are created with the VIRT keyword. VIRT is a global keyword you can use with any graphics command.

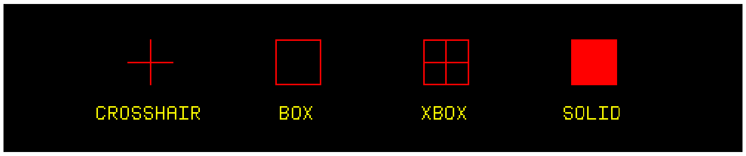
The VIRT keyword assigns a number to the virtual graphics file for identification when restoring it with the SHOWVG command. For example, VIRT=1 writes the virtual graphic to file VIRT0001. Each workstation can have up to 9999 virtual graphics.

Virtual frame files save the entire contents of the frame, unlike virtual graphics files which save only the graphics. The RVF command restores the image, graphic, frame directory, navigation, image enhancement and graphics table that were present when the frame was saved with the SVF command.

The GIF, JPG, PPM, BMP, and PostScript formats save only the pixel brightness values. These formats are useful for displaying McIDAS images on workstations not running McIDAS.

Cursor

There are 4 cursor shapes:



To change the cursor size, shape and color, use the CUR command. The size is designated by height and width and should be defined in odd numbers. If you enter even numbers, they are rounded up to odd numbers. For example, if you specify a 20 by 20 pixel cursor, the cursor will be 21 by 21 pixels. The maximum size for a cursor in McIDAS-X is display dependent; the default is 31 by 31 pixels.

Use the PC command to place the cursor at specific positions, and the E command to list the coordinates at the cursor center.

Graphics and the Cursor - Defining Graphics Parameters

In this exercise, you will use the GU (Graphics Utility) command to list the colors currently assigned to the graphics color levels, change the color assigned to a level, create new colors, save the color assignments to a graphics table, and restore a graphics table.

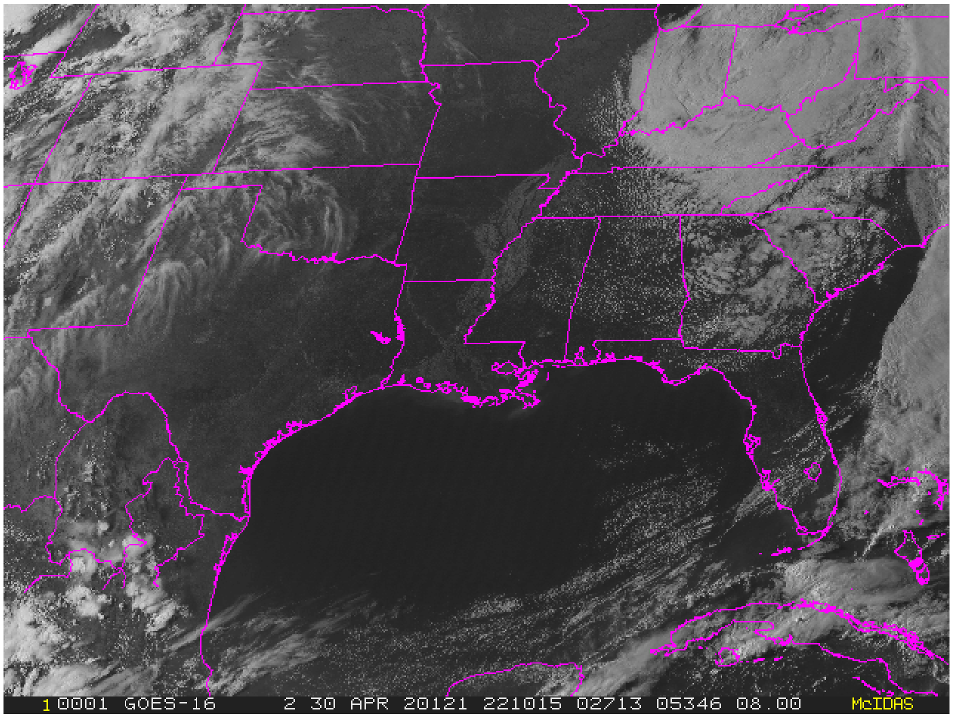
1. Start a McIDAS session.
   At the Unix prompt:
   Type: **mcidas**
   Your session should still be set for twenty frames from the last time you changed the Configuration GUI. If not, set it for twenty frames, as described in Getting Started.
2. List the colors assigned to the graphics levels in the default graphics table.
   Type: **GU TABLE**
```
GU TABLE LEVEL COLOR BLUE GREEN RED ----- ---------- ---- ----- --- 0 BLACK 0 0 0 (Graphic background) 1 MAGENTA 255 0 255 2 CYAN 255 255 0 3 YELLOW 0 255 255 4 GREEN 0 255 0 5 RED 0 0 255 6 BLUE 255 0 0 7 WHITE 255 255 255 8 GRAY 127 127 127 9 GOLD 0 187 255 10 PINK 127 127 255 11 AQUAMARINE 147 219 112 12 ORCHID 219 112 219 13 NAVY 115 0 0 14 SKY 255 163 0 15 BEIGE 127 171 255 16 PURPLE 127 0 127
```
3. Generate a map of the United States with latitude and longitude lines on frame 1; then, generate a map of the Midwest with latitude and longitude lines on frame 2. The next exercise discusses generating graphics in more detail.
   Type: **MAP USA 1 LALO 5 GRA=1;MAP MID 1 LALO 5 GRA=2**
4. List the predefined colors on your workstation.
   Type: **GU COLORS**
```
GU COLORS Colors ------------ AQUAMARINE BLACK BLUE NAVY CORAL CYAN FIREBRICK BROWN GOLD GOLDENROD GREEN GRAY GREY KHAKI MAGENTA MAROON ORANGE ORCHID PINK PLUM RED SALMON SIENNA TAN THISTLE TURQUOISE VIOLET WHEAT WHITE YELLOW BEIGE LEMON PURPLE SKY AVOCADO
```
5. Assign the color avocado to graphics color level 1, thistle to graphics color level 2, and goldenrod to graphics color level 5.
   Type: **GU MAKE 1 AVOCADO;GU MAKE 2 THISTLE;GU MAKE 5 GOLDENROD**
   Notice the graphics colors displayed in frame 1 changed when you assigned the new colors.
6. Show frame 2.
   Type: **SF 2**
   Notice that the colors on frame 2 did not change. Each frame has its own graphics table; graphics color level changes affect only that frame. However, if you specify a range of frames or a different frame than the current frame, the changes affect the graphics tables for those frames.
7. Show frame 1 and list the colors in the graphics table.
   Type: **SF 1;GU TAB**
   Notice that color levels 1, 2, and 5 have changed.
8. Create a new color by specifying the blue, green, and red intensities. Assign the new color to graphics color level 1.
   Type: **GU MAKE 1 35 107 200**
9. List the graphics table to verify that graphics color level 1 has new intensities.
   Type: **GU TABLE**
   Since you defined the color intensities, the name is blank in the table.
10. Save the graphics table to a file called NEW and list the graphics tables on your workstation to verify that it was saved.
    Type: **GU SAVE NEW;GU LIST**
    Notice that the graphics tables have a default extention of **.GRX**.
11. Restore the graphics table named GRAPHIC, which contains the default graphics color levels.
    Type: **GU REST GRAPHIC**
12. Restore the graphics table NEW to frames 1 and 2.
    Type: **GU REST NEW 1 2**
13. Verify that both frames have the new graphics table restored.
    Press: **Alt A**
    Press: **Alt B**
14. Restore the default graphics table, and erase the graphics frames. Use the GD command to change the current line width to 3 pixels.
    Type: **GU REST GRAPHIC 1 2;ERASE G 1 4;GD 3**
15. Display the second most recent GOES-16 0.5km Visible image on frame 1 centered on Raleigh-Durham, North Carolina, decrease the resolution by a factor of 8, and draw a map on the satellite image. Because of the above GD command, the line thickness level is now 3. The graphics lines will appear thicker, making the map easier to see. Note: choose different position number during the daylight if .-1 produces a black image.
    Type: **IMGDISP G16FD.-1 1 STATION=KRDU MAG=-8 BAND=2;MAP H**
16. Change the line width back to the default, 1 pixel, and erase the graphics in frame 1.
    Type: **GD 1**
    Type: **ERASE G 1**

Graphics and the Cursor - Generating Graphics

In the following two exercises, you will generate maps and create freehand drawings.

Generating Maps

1. Display the second most recent GOES-16 0.5km Visible image on frame 1 centered at Baton Rouge, Louisiana, and decrease the resolution by a factor of 8.
   Type: **IMGDISP G16FD.-1 1 STATION=KBTR MAG=-8 BAND=2 SF=Y**
2. Generate a high resolution map on the image.
   Type: **MAP VH**



3. Highlight the states of Kansas and Tennessee, by plotting the map of these states in a different color (color level 3).
   Type: **MAP X 3 STATE=KS TN**
4. Display the eighth most recent G16FD image on frame 2 centered at Baton Rouge, and generate a coastal outline map in graphics color level 5, red.
   Type: **IMGDISP G16FD.-7 2 STATION=KBTR MAG=-8 BAND=2 SF=Y;MAP L 5**
   The L parameter specifies a low resolution map conforming to the satellite image.

5. Turn the satellite image off and on so the map is easier to see.
   Press: **Alt K**
   Press: **Alt K**
6. Turn the graphics off and back on, to see the image without the map.
   Press: **Alt W**
   Press: **Alt W**
7. Erase the image in frame 3. Show frame 3 and generate a Mercator map of the USA that includes all counties in the USA.
   Type: **ERASE I 3**
   Type: **SF 3;MAP USA 3 COUNTY=ALL**
8. Display an infrared image on frame 4 and generate a map with political boundaries, coastal boundaries, and latitude and longitude lines.
   Type: **IMGDISP G16FD.-1 4 STA=KBTR MAG=-4 BAND=13 SF=Y**
   Type: **MAP X 5 LALO FILE=OUTLHPOL**
   Type: **MAP L 3**
   Notice that the political boundaries are in graphics color level 5 (red) and the coastal boundaries are in graphics color level 3 (yellow).

Adding Text and Freehand Drawings

In this exercise, you will write text to a frame with the [ZA]() command and create freehand drawings with the [ZLM]() command.

1. Show frame 1 and erase its graphics.
   Type: **SF 1;ERASE G 1**
2. Position the cursor at TV line 30 and element 100. Add the following annotation to the frame describing the image.
   Type: **PC T 30 100;ZA 5 10 "GOES VISIBLE FOR (DAY) AT (TIME) UTC**

The phrase GOES VISIBLE FOR (the image day) AT (the image time) UTC is written on frame 1. The first parameter of the ZA command (5) indicates the graphics color level; the second parameter (10) indicates the height of the letters in pixels.

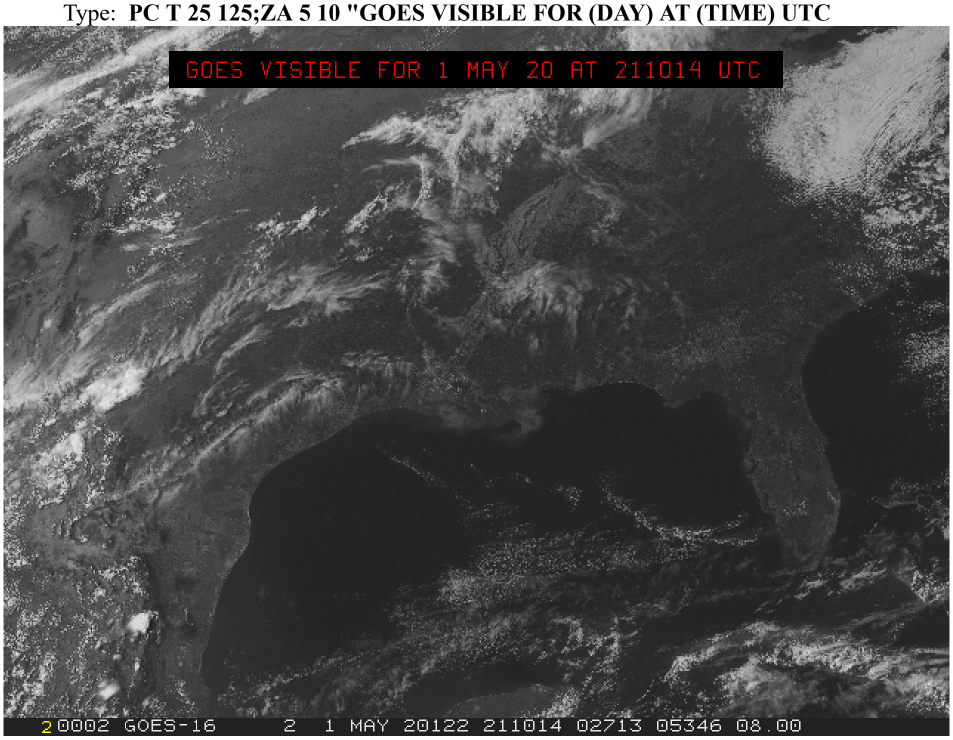
3. Show frame 2 and erase both the image and the graphics.
   Type: **SF 2;ERASE**
4. Activate the freehand drawing command ZLM.
   Type: **ZLM DRAW**
5. Press and hold the middle mouse button and draw something on the screen by moving the mouse.

6. End the draw option by pressing the right and middle mouse buttons simultaneously, or:
   Press: **Alt Q**

Graphics and the Cursor - Erasing Graphics

In this exercise, you will erase the graphics inside the cursor with the CW command and then erase all the graphics inside the current frame with the ERASE command.

1. Show frame 1.
   Type: **SF 1**
2. Place the cursor at TV line 30 and element 320. Erase all the graphics inside a 25 by 440 cursor and set the color in the box to graphic level 8. Use an X for the frame number parameter.
   Type: **PC T 30 320;CW 8 X 25 440**
   The graphics and image inside the 25 by 440 cursor are restored to the color specified (gray).

3. Change graphics color level 8 to black.
   Type: **GU MAKE 8 BLACK**
4. Position the cursor at TV line 25 and element 125, and add the annotation again. The black background makes it easier to read.
   Type: **PC T 25 125;ZA 5 10 "GOES VISIBLE FOR (DAY) AT (TIME) UTC**
   

5. Erase the graphics in frames 1 through 4.
   Type: **ERASE G 1 4**
   The graphics are cleared, leaving the frames either empty or with a satellite image.

6. Restore the default graphics table to frames 1 through 4.
   Type: **GU REST X 1 4**

Graphics and the Cursor - Saving and Restoring Graphics and Frames

In this exercise, you will save and restore virtual graphics files, virtual frame files, and GIF files.

Virtual Graphics Files

In these five steps, you will generate a map over a satellite image and save the map as a virtual graphics file with the VIRT keyword. Then you'll erase the image and the graphics, locate the saved virtual graphics file, and restore it with the SHOWVG command.

1. Show frame 1.
   Type: **SF 1**
2. Generate a map over the image and save it as a virtual graphic assigned to file 9110. You must assign the graphic to a virtual file at the time you run the graphics command.
   Type: **MAP H VIRT=9110**
   The VIRT keyword is a global keyword which displays a specified virtual graphic. This keyword only works with commands that generate graphics. The output is written to virtual graphics file VIRT9110 as well as the frame.

3. List virtual graphic 9110 on your workstation to verify it was saved.
   Type: **DMAP VIRT9110**
   ```
   DMAP VIRT9110 PERM SIZE LAST CHANGED FILENAME DIRECTORY ---- --------- ------------ -------- --------- -rw- 69772
   Oct 26 16:06 VIRT9110 /home/user/mcidas/data 69772 bytes in 1 files
   ```
4. Erase the image and graphics.
   Type: **ERASE**
5. Restore virtual graphic 9110 to frame 1.
   Type: **SHOWVG 9110 1**
   Notice only the graphic was saved not the satellite image.

6. Erase the graphic on frame 1.
   Type: **ERASE G**

Virtual Frame Files

In the next five steps, you will use the SVF command to save a satellite image and map. Then you'll list the virtual frame files on your workstation using the LVF command, and restore a virtual frame file using the RVF command.

1. Display the second most recent GOES-16 0.5km Vis image centered on Washington DC, draw a map, and set the loop bounds from 1 to 4.
   Type: **IMGDISP G16FD.-1 1 STA=KDCA MAG=-4 BAND=2; MAP H;LS 1-4**
2. Save the satellite image and map as a virtual frame file named SATMAP.
   Type: **SVF 1 X SATMAP**
   Be sure the message "Frames saved in SATMAP.PIX - 1" is displayed in the text window before continuing.

3. List the virtual frame files on your workstation.
   Type: **LVF**
   A list of virtual frame files and corresponding information for each is displayed in the text window as shown below. Notice that virtual frame files are saved with the extension .PIX.

   ```
   PERM SIZE LAST CHANGED FILENAME DIRECTORY ---- --------- ------------ ---------- --------- -rw- 381104 Oct 26 10:16
   SATMAP.PIX /home/user/mcidas/data 381104 bytes in 1 files
   ```
4. List information about the number of frames, frame size, graphics, and image levels for the virtual frame file SATMAP. The frame size and graphic and image levels in the virtual frame file must be the same as the frame where it is being restored.
   Type: **LVF SATMAP FORM=ALL**
   ```
   PERM SIZE LAST CHANGED FILENAME DIRECTORY ---- --------- ------------ ---------- --------- -rw- 381104 Oct 26 10:16
   SATMAP.PIX /home/user/mcidas/data 381104 bytes in 1 files file is in new format and contains 1 frames frame 1 is 480
   by 640 8-bit LVF: DONE
   ```
5. Show frame 2 and restore the virtual frame file you saved as SATMAP.
   Type: **SF 2;RVF 2 X SATMAP**
   Press: **Alt B**
   Press: **Alt A**
   The image and graphics are displayed in frame 2. Notice that the frame's annotation line is saved with the image and may not match the frame number where the virtual frame file is restored. When you advance through the frames, use the status line to determine which frame is displayed.

GIF files

In the following steps, you will load an area into frame 4, draw a map on it and then save the image as a GIF file.

1. Display the second most recent GOES-16 1km Vis image centered on Washington DC onto frame 4 and draw a map on the image.

Type: **IMGDISP G16FD.-1 4 STA=KDCA MAG=-2 BAND=3 SF=Y; MAP H**

2. Save frame 4 in GIF format.

   Type: **FRMSAVE 4 PICTURE**

   The default for FRMSAVE is to save the frame in the GIF format, but you can also specify the FORM keyword to choose to save it in a PPM, BMP, JPEG, PostScript, or color PostScript format.

3. List the GIF image PICTURE on your workstation to verify that it was saved and to where it was saved.

   Type: **DMAP PICTURE**

   Notice that the file was saved with the extension .GIF

4. Use your workstation's image viewer (e.g. display, xloadimage, xv) to display the GIF file from a terminal window.

Graphics and the Cursor - Changing the Cursor Shape, Color, and Size

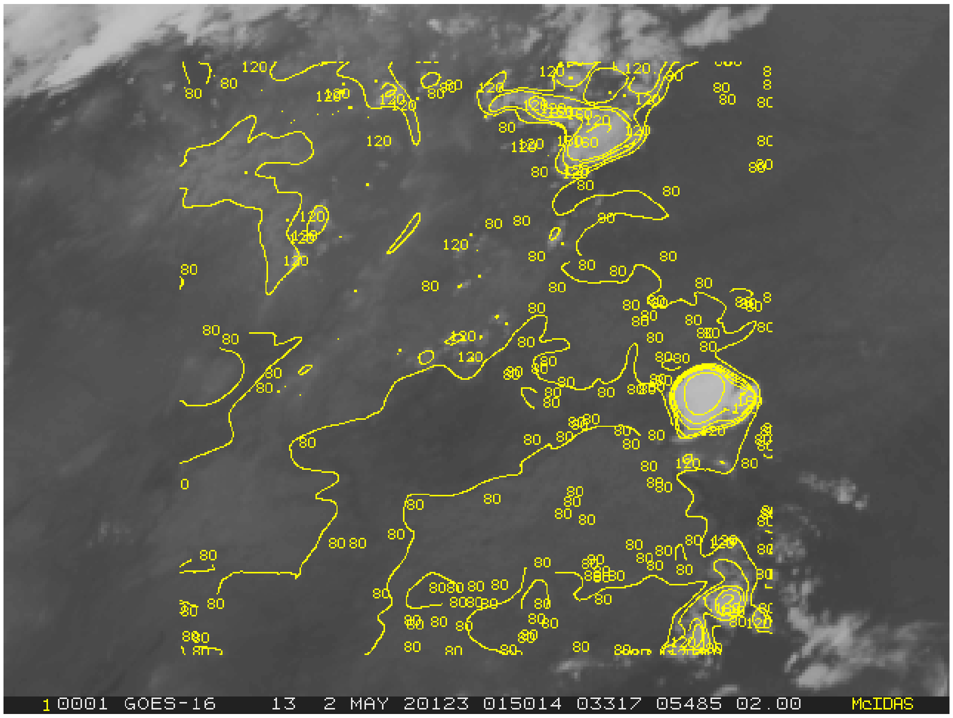
In this exercise, you will use the CUR command to change the cursor size, shape, and color.

1. Define the cursor size as 31 by 63 pixels, the color as blue, and the type as XBOX. Move the cursor onto the image window.
   Type: **CUR 31 63 XBOX BLUE**
2. Define the cursor as a 25 by 25 pixel solid cursor in a custom color.
   Type: **CUR 25 25 SOLID 63 128 255**
3. Change the cursor to its default setting.
   Type: **CUR**

Graphics and the Cursor - Contouring Data With the Cursor

In this exercise, you will contour the temperatures and brightness levels of an image.

1. Erase the graphics in frame 1. Display the second most recent GOES-16 2km IR image on frame 1 centered on 36° N and 104° W. Magnify the resolution by a factor of 2.
   Type: **ERASE G 1;IMGDISP G16FD.-1 1 LATLON=36 104 MAG=2 BAND=13 SF=YES**
2. Position the cursor in the center of the frame. Contour the brightness levels using a contour interval of 20 inside a 401 by 401 pixel cursor.
   Type: **PC C;IMGPROBE CONT BOX BRIT SIZE=401 401 CINT=20 MODE=N**



3. Display the second most recent GOES-16 2km IR image on frame 2 centered on 36° N and 104° W. Magnify the resolution by a factor of 2.
   Type: **IMGDISP G16FD.-1 2 LATLON=36 104 MAG=2 BAND=13 SF=Y**
4. Position the cursor in the center of the frame. Contour the temperature inside a 401 by 401 pixel cursor. Specify a contour interval of 5 Kelvin.
   Type: **PC C;IMGPROBE CONT BOX TEMP SIZE=401 401 CINT=5 MODE=N**

Graphics and the Cursor - Zooming with the Cursor

In this exercise, you will zoom in on features of an image, and use the roam capabilities of McIDAS.

1. Erase the graphics in frames 1 and 2. Display the second most recent GOES-16 2km IR image on frame 1 using a magnification of MAG=-3.
   Type: **ERASE G 1 2;IMGDISP G16C.-1 1 BAND=13 MAG=-3 SF=YES**
2. Set the zoom factor to 3, and zoom in on the image.
   Type: **ZM 3**
   Press: **Alt Z**
3. Press and hold the middle mouse button to roam around the image, looking at the interesting features of the current weather.

   If you specified the `-autoResize` flag in your `.mcidasrc` file, the Image Window is automatically resized to fit the zoomed frame. If you did not specify this flag, alt-R will resize your window.
4. Toggle the zoom function off.
   Press: **Alt Z**

Graphics and the Cursor - Locating Specific Positions Using the Cursor

In this exercise, you will use the STNLIST command to find the station ID for Morgantown, West Virginia, position the cursor at that location using the PC command, display its earth coordinates using Alt E, and list the towns closest to Morgantown with the TOWNS command. Then you'll follow the same procedure for Montreal, Canada using the S command instead of **TOWNS**. You will also use the STNPLOT command to plot all of the RAOB stations on your image.

1. Display a recent GOES-16 visible image on frame 2 centered over Bangor, Maine. Add a high resolution map.
   Type: **IMGDISP G16FD.-1 2 BAND=13 STA=KBGR MAG=-2 SF=Y;MAP H**
2. List all of the METAR stations in West Virginia.
   Type: **STNLIST ST=WV TYPE=M**
   The METAR stations in West Virginia are listed, including their station number, station ID, data types, state and country, earth coordinates, and elevation, as shown below.

```
STNLIST ST=WV TYPE=M IDN ID Station Name Data Types ST CO LAT LON ELE ----- ----- --------------------
----------------- -- -- ---------- ---------- ---- --- K3I2 Point Pleasant M EG WV US 38:54:52 82:05:54 196 --- K48I
Sutton M EG WV US 38:41:13 80:39:06 387 --- K6L4 LOGAN M WV US 37:51:00 81:55:00 508 72412 KBKW RALEIGH COUNTY MEM A
M 36 T EG WV US 37:48:00 81:07:00 763 --- KBLF BLUEFIELD MERCER COU M T EG WV US 37:18:00 81:12:00 871 --- KCKB
HARRISON/MARION RGNL M T EG WV US 39:18:00 80:13:00 367 72414 KCRW YEAGER AIRPORT M 36 T EG WV US 38:22:00 81:36:00
299 72417 KEKN JENNINGS RANDOLPH FI M 36 T EG WV US 38:53:00 79:51:00 606 --- KHLG WHEELING OHIO COUNTY M T EG WV US
40:10:00 80:39:00 364 72425 KHTS TRI-STATE AIRPORT M 36 T EG WV US 38:22:00 82:33:00 252 --- KI16 Pineville M EG WV
US 37:36:01 81:33:33 544 --- KLWB Lewisburg / Greenbri M T EG WV US 37:52:00 80:24:00 702 --- KMGW MORGANTOWN MUNI-
HART M T EG WV US 39:39:00 79:55:00 380 --- KMRB MARTINSBURG EASTERN M T EG WV US 39:24:00 77:59:00 170 --- KPKB
MID-OHIO VALLEY RGNL M T EG WV US 39:20:00 81:27:00 262 72427 KPKB PARKERSBURG M 36 T EG WV US 39:21:00 81:25:00 262
--- KRLX Charleston M WV US 38:19:00 81:43:00 364 --- KSSU White Sulphur Sprng M WV US 37:46:00 80:20:00 549 ---
KW22 Buckhannon M EG WV US 39:00:00 80:16:00 498 --- KW99 Petersburg M EG WV US 38:59:00 79:08:00 293 Number of
stations listed: 20 STNLIST: Done
```

3. List the stations in West Virginia that include MORGANTOWN in their name.
   Type: **STNLIST ST=WV MATCH=MORGANTOWN**

```
STNLIST ST=WV MATCH=MORGANTOWN IDN ID Station Name Data Types ST CO LAT LON ELE ----- ----- --------------------
----------------- -- -- ---------- ---------- ---- --- BBO MORGANTOWN V WV US 39:41:00 79:56:00 0 --- KMGW
MORGANTOWN MUNI-HART M T EG WV US 39:39:00 79:55:00 380 --- MGW MORGANTOWN MUNI-HART V F WV US 39:39:00 79:55:00 380
Number of stations listed: 3 STNLIST: Done
```

4. Position the cursor over Morgantown.
   Type: **PC L KMGW**

   The PC command places the cursor over the designated location (L indicates location) on the displayed frame.

5. Display the coordinates at the cursor position.
   Press: **Alt E**
   Notice that the latitude and longitude are the same or very close to those listed in the STNLIST output.

```
Frame Latitude Longitude Tvline Tvelem Line Elem 2 39:39:43 079:56:18 329 104 3117 10041
```

6. List the towns closest to Morgantown and then press Alt-Q to quit the TOWNS command.
   Type: **PC L KMGW;TOWNS**
   Press: **Alt G**
   Press: **Alt Q**
7. If you would like to enter the current lat/lon position of your cursor onto the command, you can press the Ctrl button on your keyboard and the left mouse button simultaneously. Enter a lat/lon pair into an IMGPROBE command using this method.
   Type: **IMGPROBE MODE=N LATLON=** - but don't press Enter!
   Press: **Ctrl LeftMouseButton**
   Press: **Enter**

   This should give you the image information for your current cursor location.

8. Find the country code for Canada, as you will need it for the CO keyword in step 9.
   Type: **CCODE CANADA**
9. Find the station code for Montreal.
   Type: **STNLIST CO=CA MATCH=MONTREAL**
   The text window lists the stations that include Montreal in their name, as shown below. Use the 4-letter station code for Montreal/Mirabel.

```
STNLIST CO=CA MATCH=MONTREAL IDN ID Station Name Data Types ST CO LAT LON ELE ----- ----- --------------------
----------------- -- -- ---------- ---------- ---- --- CWGJ Montreal River Radar M CA 47:14:00 84:31:00 559 --- CZGJ
MONTREAL RIV M ON CA 47:15:00 84:36:00 520 71675 CWPQ Montreal-Est M 36 QB CA 45:38:00 70:33:00 51 --- CYMX Montreal
/ Mirabel M T QB CA 45:41:00 74:02:00 82 71627 CYUL Montreal / Pierre El M 36 T QB CA 45:28:00 73:45:00 31 --- IDO
MONTREAL V QB CA 45:28:00 73:45:00 0 --- UTM ST MATHIAS/MONTREAL V QB CA 45:20:00 73:16:00 0 --- WAO MONTREAL V QB
CA 45:28:00 73:45:00 0 --- YMX MONTREAL V QB CA 45:50:00 74:23:00 82 --- YUL MONTREAL P.Q. V QB CA 45:31:00 73:58:00
36 71183 CWTQ MONTREAL/DORVAL M 36 QC CA 45:28:00 73:45:00 32 71626 CYMX MONTREAL/MIRABEL M 36 T QC CA 45:40:00
74:01:00 82 --- WPQ MONTREAL EST V QC CA 45:38:00 73:30:00 6 Number of stations listed: 13 STNLIST: Done
```

10. Position the cursor at Montreal.
    Type: **PC L CYMX**
11. Display the coordinates at the cursor position.

Press: **Alt E**

12. List the METAR, Synoptic and RAOB stations that are closest to Montreal.
    Type: **PC L CYMX**
    Press: **Alt S**
13. Plot all of the RAOB stations that are within the navigation of your image frame.
    Type: **STNPLOT TYPE=R**

Graphics and the Cursor - Finding Distances Between Points

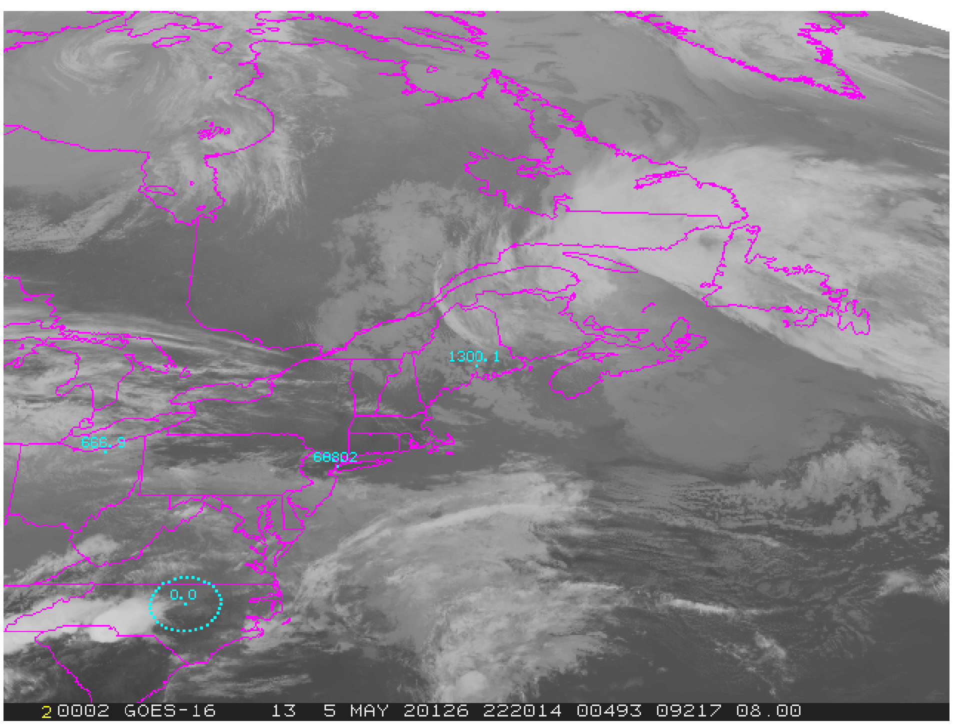
In this exercise, you will use the cursor and the DIST command to find the distance between points on a navigated frame. When using the DIST command, it is useful to type the commands while the image window is raised.

1. Click in the image window to raise it to the foreground.

2. Position the cursor at Raleigh-Durham, North Carolina and start the DIST command. Press Alt-G to enter Raleigh-Durham as the starting point for measuring the distances.
   Type: **PC L KRDU;DIST**
   Press: **Alt G**
3. Position the cursor at Bangor, Maine. Then press the middle mouse button or Alt-G to measure the distance from Raleigh-Durham to Bangor.
   Type: **PC L KBGR**
   Press: **Alt G**
   A distance of 1300.1 kilometers is marked on the frame and also listed in the text window with the latitude and longitude measurements.

4. Position the cursor at New York City. Then press Alt-G to measure the distance from New York City to the starting point (Raleigh-Durham, 688.2 km) and to the previous point (Bangor, 612.2 km). Until you reset the starting point, all of the starting point measurements will be made from Raleigh-Durham.
   Type: **PC L KNYC**
   Press: **Alt G**
5. Press the right mouse button to clear the starting point. Then position the cursor at New York City and press the middle mouse button or Alt-G to set New York City as the new starting point.
   Press: **< right mouse button >**
   Type: **PC L KNYC**
   Press: **Alt G**
6. Position the cursor at Cleveland, Ohio, and press the middle mouse button or Alt-G to measure the distance from New York City to Cleveland.
   Type: **PC L KCLE**
   Press: **Alt G**
   The distance from New York City to Cleveland (666.9 km) is calculated.

7. End the **DIST** command by pressing the middle and right mouse buttons, or:
   Press: **Alt Q**
8. Position the cursor at Raleigh-Durham and draw a circle encompassing the area within 100 km of the point.
   Type: **PC L KRDU;DIST CIR X X 100**
   This is useful for quickly determining the approximate distance between a weather system and a particular location, as shown below.



9. Exit McIDAS.
   Type: **EXIT**

Enhancements

Table of Contents:

Enhancements

Enhancements are used to improve image contrast and produce colored imagery. This allows you to emphasize the image features you are analyzing. For example, you can enhance thunderstorm cloud tops, low clouds, or fog.

This lesson describes how to:

- create color enhancements
- change grayscale contrast

The following commands are used in this lesson.

Command	Function
BAR	draws a grayscale bar on a frame containing an image
EB	black and white contrast stretching
EU	enhancement utility
IMGDISP	displays image data
SU	image data stretching utility

Enhancements - Basic Concepts

There are two types of enhancements: color enhancements and grayscale enhancements. A color enhancement changes grayshades to colors and a grayscale enhancement changes a grayshade to a different grayshade value.

Color Enhancements

A color enhancement is a table of colors that corresponds to brightness values. Color enhancements are useful for tracking cloud features. For example, to track the tops of thunderstorms overshooting the tropopause, you can color all brightness values between 180 and 250 red.

Color enhancements are created with the EU MAKE command. You can create a color enhancement by assigning a color to a brightness value or a brightness range. For example, you could create an enhancement table where the color green corresponds to the brightness range 50 to 79, the color blue corresponds to the brightness range 80 to 99, and the color red corresponds to brightness value 100.

In addition, you can create a color enhancement by specifying color intensities. The values within the brightness range are interpolated within the color intensity range. For example, if the brightness range 0 to 71 is assigned to a blue color intensity of 203 to 255, a green color intensity of 173 to 200, and a red color intensity of 3 to 100, as shown below, the pixels with a low brightness value (near 0) will have corresponding low red, green, and blue intensities, and the pixels with high brightness values (near 71) will have corresponding high red, green, and blue intensities.

```
Brightness Blue Green Red min max min max min max min max --- --- --- --- --- --- --- --- 0 71 203 255 173 200 3 100
```

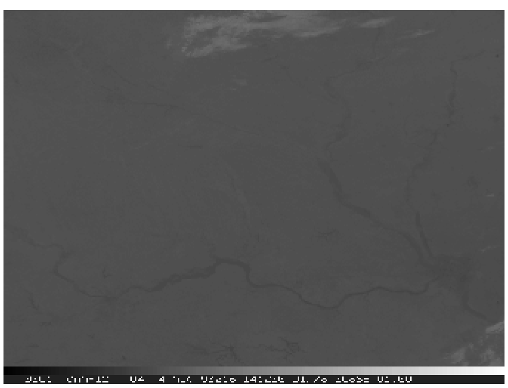
Once you create an enhancement table, you can save it using EU SAVE and then restore it using the command EU REST. You can apply the same or different enhancement tables to each frame on the workstation.

Grayscale Enhancements

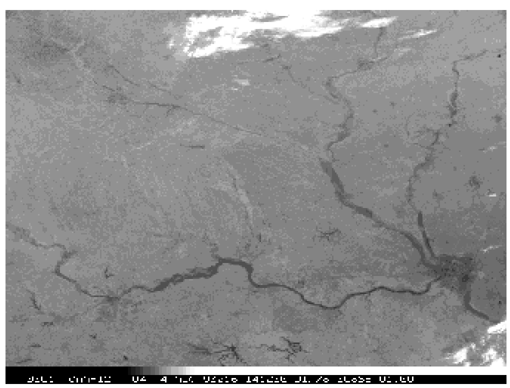
Normally, a pixel's digital value, stored in an area, correlates to a brightness value. Each brightness value appears as a different shade of gray when the image is displayed. When a grayscale enhancement is applied, the correlation between the digital values and the displayed grayshades changes. You can change the grayscale contrast of an image two ways: using image contrast stretching or using image data stretching.

Image Contrast Stretching

Image contrast stretching changes the grayscale of the displayed image; it does not change the area data values. You can change the grayscale contrast of an image using the EB command. You can run the EB command two ways: using the command line and using the mouse. Using the command line, you specify the lower and upper brightness values to be enhanced. All pixels with brightness values below the lower input values and above the upper input value remain unchanged. The brightness values between the range are linearly interpolated. Using the mouse controlled version, you move the mouse to increase or decrease the brightness of the image. You can save grayscale enhancements and apply them to other images using the EU SAVE and EU REST commands. The example below shows the original contrast of an image and the contrast of an image after contrast stretching.



Original Image



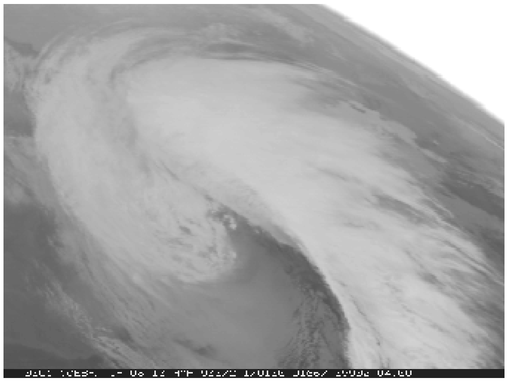
Contrast stretched image

Image Data Stretching

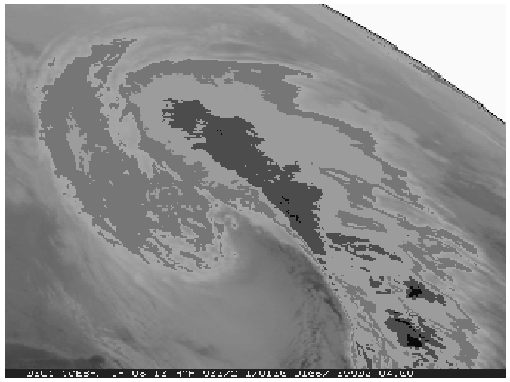
Image data stretching changes the grayscale of an image by stretching area data values to brightness values. To stretch the image data values, you must create a table that defines the values to stretch, as shown below.

```
SU TABLE MB BREAKPOINTS STORED IN TABLE : MB.ST INPUT OUTPUT ----- ------ 162.8 250 192.3 250 192.4 250 209.3 10 209.4 10
213.3 10 213.4 75 219.3 75 219.4 156 230.3 156 230.4 117 241.3 117 241.4 167 279.8 102 279.9 102 301.9 0 302 0 330 0
CALIBRATION TYPE : AAA CALIBRATION UNITS : TEMP BAND NUMBER : -1 INTERPOLATION TYPE: LIN
```

The SU command defines tables to stretch raw, radiance, temperature, albedo, or brightness values (depending on the calibration type) to a user-defined brightness value. Stretch tables are used with the IMGDISP command to emphasize weather features in an image. The example below shows an image before and after an MB data stretch table was applied.



Original Image



Data Stretched image

Enhancements - Creating Color Enhancements

In this exercise, you will use the EU command to assign colors to brightness values. You will create a simple enhancement that assigns brightness values to colors and another enhancement that assigns various brightness ranges to color ranges.

1. Start a McIDAS session.
   At the Unix prompt:
   Type: `mcidas`
   Your session should still be set for twenty frames from the last time you changed the Configuration GUI. If not, set it for twenty frames, as described in Getting Started.
2. Set the loop sequence to frames 1 through 6.
   Type: **LS 1-6**
3. Display the second most recent GOES-16 2km IR image on frame 1 centered on New Orleans. Decrease the image resolution by a factor of 4 and place a grayscale bar on the frame.
   Type: **IMGDISP G16C.-1 1 STA=KNEW BAND=13 MAG=-4 SF=YES;BAR LINT=10**
4. Assign the color red to the brightness range 180 to 220 to color enhance the cloud tops.
   Type: **EU MAKE 180 220 RED**

   Notice how the grayscale bar also changes to reflect the changes to the enhancement.

5. List the enhancement table. The brightness values between 180 and 220 are assigned to a red intensity range of 255 to 255.
   Type: **EU TABLE**
```
Brightness Blue Green Red min max min max min max min max --- --- --- --- --- --- --- --- 0 179 0 179 0 179 0 179
180 220 0 0 0 0 255 255 221 255 221 255 221 255 221 255
```
6. Assign the brightness range 180 to 220 to the range of colors between yellow and red.
   Type: **EU MAKE 180 220 YELLOW RED**
   The brightness value 180 is yellow, the brightness value 220 is red, and the values in between are displayed as various shades between the two colors.

7. List the enhancement table.
   Type: **EU TABLE**
   The enhancement table is listed as shown below. The blue, green, and red intensity values of 0, 255, and 255 create the color yellow and are assigned to brightness value 180. The blue, green, and red intensity values of 0, 0, and 255 create the color red and are assigned to brightness value 220.
```
Brightness Blue Green Red min max min max min max min max --- --- --- --- --- --- --- --- 0 179 0 179 0 179 0 179
180 220 0 0 255 0 255 255 221 255 221 255 221 255 221 255
```
8. Save the enhancements with the name STORM.
   Type: **EU SAVE STORM**
9. Restore the default enhancement to the frame.
   Type: **EU REST**
10. Restore the enhancement STORM to the frame.
    Type: **EU REST STORM**
11. List the enhancement tables in your account on the workstation.
    Type: **EU LIST**
    The file STORM.ET should be listed.

12. Delete the enhancement STORM.
    Type: **EU DEL STORM**
13. Restore the default enhancement to frames 1 through 6.
    Type: **EU REST X 1 6**
    The X after the REST parameter indicates to use the default enhancement.

14. Erase the image and the graphics from frame 1.
    Type: **ERASE**

Enhancements - Changing Grayscale Contrast

In this section, you will use two methods to create a grayscale contrast. First, you will create image contrast stretching and then you will create and apply image data stretching.

Creating Image Contrast Stretching

In this exercise, you will use the EB command to change the grayscale contrast of an image. First, you will use the mouse to move the cursor over the image and stretch the grayscale contrast. Then, you will input the values manually.

1. Display the first GOES-16 visible image of Hurrican Dorian on frame 1 centered on 28° N and 79° W, and add a gray scale bar.
   Type: **IMGDISP DORIAN/M1.1 1 LATLON=28 79 GRAY=YES**
2. Initiate mouse-controlled grayscale stretching.
   Type: **EB**
3. Move the cursor to the image window.

4. Move the mouse to the right to brighten the image. The range of pixels with a brightness near 255 (white) increases, as shown in the gray scale bar at the bottom of the frame.

5. Move the cursor towards the top of the frame to decrease the image brightness. The range of pixels with a brightness near 0 (black) increases, as shown in the gray scale bar at the bottom of the frame.

6. Find an enhancement that you like and press the right mouse button to end the enhancement.

7. Save the grayscale enhancement as GRAY.
   Type: **EU SAVE GRAY**
8. List the brightness value at the center of the image.
   Type: **PC C**
   Press: **Alt D**
9. Restore the original grayscale of the image.
   Type: **EU REST**
10. List the brightness value at the center of the image.
    Type: **PC C**
    Press: **Alt D**
    Note the values are the same as those in step 8. IMGPROBE (Alt-D) does not list the values that were modified with image contrast stretching, but lists the values stored in the area.

Now, you will manually input the brightness values with the EB command.

1. To get a stretch of the brighter values of the hurricane, find a pair of BRIT values to stretch between. Position the cursor at TV coordinates (268,364) and (389,319) to find the brightness values.
   Type: **PC T 268 364;IMGPROBE MODE=N**
   Type: **PC T 389 319;IMGPROBE MODE=N**
   The brightness values are 245 and 194.

2. Rescale the brightness values 194 to 245 to go from 0 to 255. Brightness value 194 will become 0 and value 245 will become 255. All values in between will be linearly stretched between 0 and 255.
   Type: **EB 194 245 0 255**
   Since most of the brightness values of the hurricane are between 194 and 245, creating an enhancement for this range makes the image features more prominent. Note that all brightness values outside the range of 194 to 245 remain unchanged.

3. Save the enhancement as GRAY2.
   Type: **EU SAVE GRAY2**
4. Restore the default enhancement table to the frame.
   Type: **EU REST**
5. List the enhancement tables that start with GRAY on your workstation.
   Type: **EU LIST GRAY**
   ```
   PERM SIZE LAST CHANGED FILENAME DIRECTORY ---- --------- ------------ -------- --------- -rw- 3268 Oct 29 17:01
   GRAY.ET /home/user/mcidas/data -rw- 3268 Oct 29 17:01 GRAY2.ET /home/user/mcidas/data 6536 bytes in 2 files
   ```
6. Delete the saved enhancement tables.
   Type: **EU DEL GRAY;EU DEL GRAY2**

Creating and Applying Image Data Stretching

Next, you will define stretch tables to stretch brightness and temperature values stored in an area. Then, you will apply the stretch tables to images and compare the stretched values to the original values.

1. Display a local GOES-16 0.5km Visible image on frame 1 centered on New Orleans. Decrease the resolution by a factor of 8 and add a high

resolution map.
    Type: **IMGDISP ABI/CONUS 1 STA=KNEW MAG=-8 BAND=2; MAP H**

2. Position the cursor at the center of the frame and list the brightness value.
    Type: **PC C**
    Press: **Alt D**

   The brightness value at the center is 67.

3. Next, initialize a stretch table named LEARN to stretch brightness values. The VISR parameter specifies the data type as GOES 1-byte data.
    Type: **SU INI LEARN VISR BRIT**
4. Define the brightness ranges to stretch. Assign the brightness value 0 to 255 and the value 255 to 0 to make light areas dark and dark areas light.
    Type: **SU MAKE LEARN 0 255 255 0**
5. List the breakpoints in the stretch table.
    Type: **SU TABLE LEARN**

   The table lists the brightness values and the corresponding stretched values as shown below.

```
SU TABLE LEARN BREAKPOINTS STORED IN TABLE : LEARN.ST INPUT OUTPUT ----- ------ 0 255 255 0 CALIBRATION TYPE : VISR CALIBRATION UNITS : BRIT BAND NUMBER : -1 INTERPOLATION TYPE: LIN SU: DONE
```

6. Display a local GOES-16 0.5km Visible image on frame 2 centered on New Orleans. Decrease the resolution by a factor of 8, apply the stretch table LEARN, and add a high resolution map.
    Type: **IMGDISP ABI/CONUS 2 STA=KNEW MAG=-8 BAND=2 SU=LEARN SF=YES;MAP H**
7. Set the loop bounds from 1 to 2 and compare the images.
    Type: **LS 1-2**
    Press: **Alt A**
    Press: **Alt B**
8. Show frame 2 and list the areas values at the cursor's center.
    Type: **SF 2;PC C;IMGPROBE MODE=N**

   Notice that there is a MODB/LEARN data type listed in the output of the D command, as shown below. This lists the value of the stretched data.

```
Image Name Day Nominal Time Scan Time Band ---------------- ------- ------------ --------- ---- ABI/CONUS.1 4 Jan 18004 16:57:20 16:58:24 2 File Nominal Image RAW RAD ALB BRIT MODB Lat/Lon Line/Element Line/Element * % LEARN 30:03:14/ 90:02:11 3015/ 4470 3016/ 4471 362 37.121 6.91 67 188 * watts/meter**2/steradian/micron
```

   Since the values in the table are reversed (0 is now 255 and 255 is now 0), you can calculate the stretched value of a pixel by subtracting the pixel's original brightness value from the maximum value. For example, to calculate the stretched value of the center pixel, subtract the original brightness value (67) from the maximum brightness value (255); the stretched value of the center pixel is 188 (255-67).

9. List the stretch tables on the workstation.
    Type: **SU LIST**

Next, you will create a Multiple Breakpoint (MB) stretch table to enhance clouds in a GOES infrared image and create an approximate MB stretch curve.

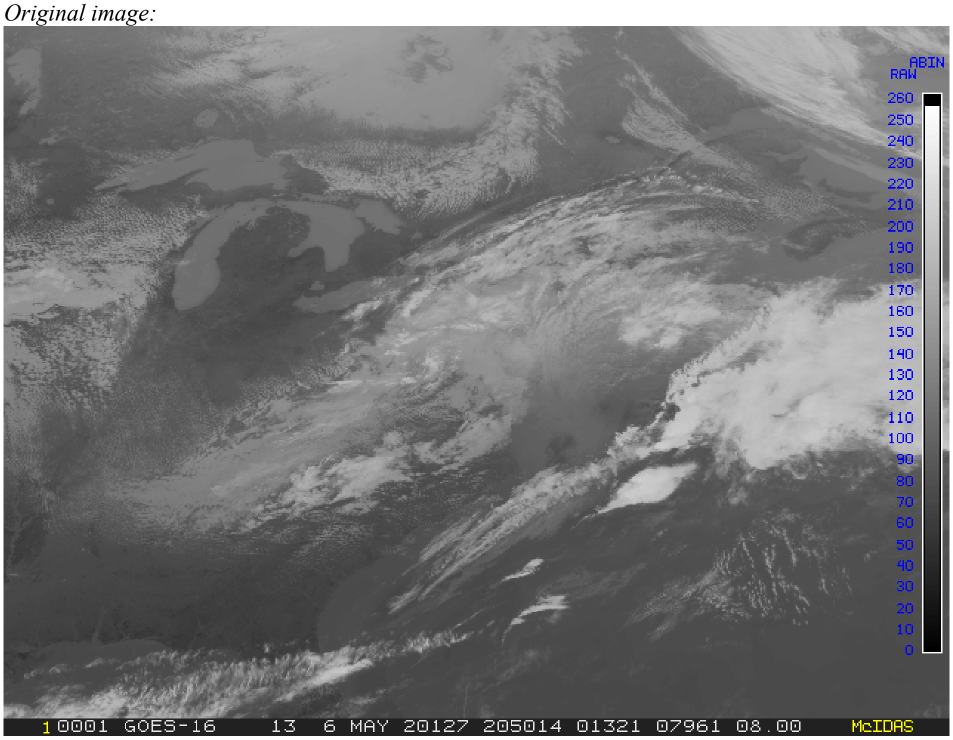
1. Erase the graphics on frames 1 and 2. Display the second most recent GOES-16 2km IR image on frame 1 centered on Washington, DC, reduce the resolution by 2 and show the frame. Use the REFRESH keyword of [IMGDISP] to draw a data bar on the image and label every 10th value in blue (color 6).
    Type: **ERASE G 1 2;IMGDISP G16FD.-1 1 STA=KDCA MAG=-2 BAND=13 SF=YES REFRESH='BAR (GRA) LINT=10 COLOR=6'**
2. Initialize a stretch table named MB to stretch temperature values to a brightness range. The X parameter uses a default for any type of data.
    Type: **SU INI MB X TEMP**
3. Assign the temperature values between 330° K and 302° K to the brightness value 0 in the MB stretch table.
    Type: **SU MAKE MB 330 302 0 0**
4. Assign the temperature values between 301.9° K and 279.9° K to the brightness range 0 to 102 in the MB stretch table.
    Type: **SU MAKE MB 301.9 279.9 0 102**
5. Assign the temperature values between 279.8° K and 241.4° K to the brightness range 102 to 167 in the MB stretch table.
    Type: **SU MAKE MB 279.8 241.4 102 167**
6. Assign the temperature values between 241.3° K and 230.4° K to the brightness value 117 in the MB stretch table.
    Type: **SU MAKE MB 241.3 230.4 117 117**
7. Assign the temperature values between 230.3° K and 219.4° K to the brightness value 156 in the MB stretch table.
    Type: **SU MAKE MB 230.3 219.4 156 156**
8. Assign the temperature values between 219.3° K and 213.4° K to the brightness value 75 in the MB stretch table.
    Type: **SU MAKE MB 219.3 213.4 75 75**
9. Assign the temperature values between 213.3° K and 209.4° K to the brightness value 10 in the MB stretch table.
    Type: **SU MAKE MB 213.3 209.4 10 10**
10. Assign the temperature values between 209.3° K and 192.4° K to the brightness range 10 to 250 in the MB stretch table.
    Type: **SU MAKE MB 209.3 192.4 10 250**
11. Assign the temperature values between 192.3° K and 162.8° K to the brightness value 250 in the MB stretch table.
    Type: **SU MAKE MB 192.3 162.8 250 250**
12. Verify that the stretch table contains the correct breakpoints.
    Type: **SU TABLE MB**

The table MB is listed as shown below.

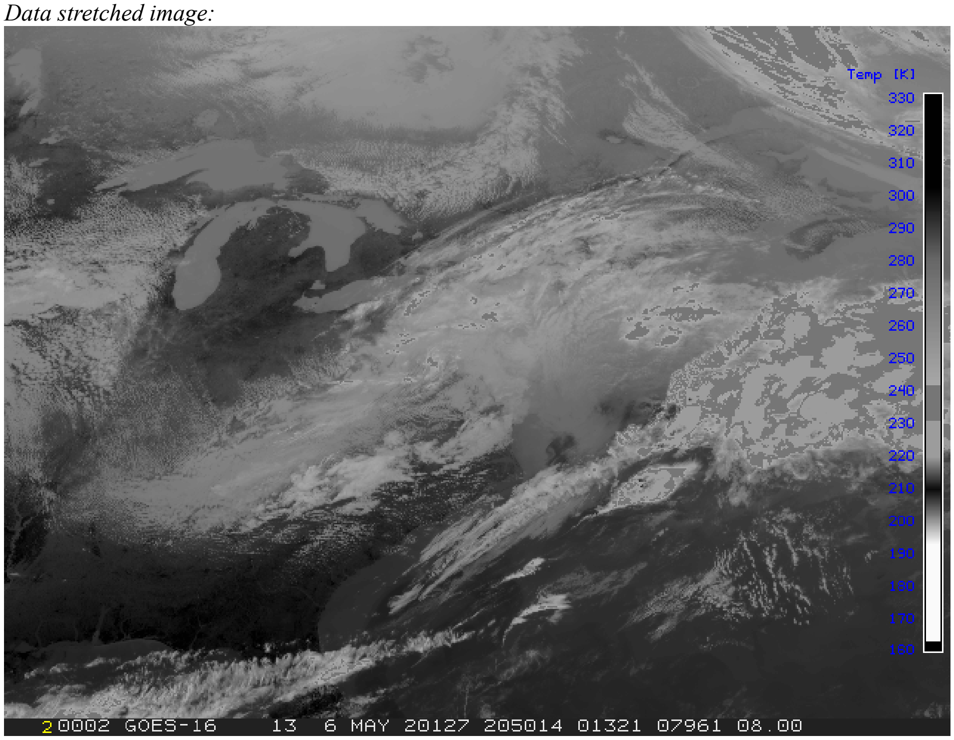
```
SU TABLE MB BREAKPOINTS STORED IN TABLE : MB.ST INPUT OUTPUT ----- ------ 162.8 250 192.3 250 192.4 250 209.3 10
209.4 10 213.3 10 213.4 75 219.3 75 219.4 156 230.3 156 230.4 117 241.3 117 241.4 167 279.8 102 279.9 102 301.9 0
302 0 330 0 CALIBRATION TYPE : CALIBRATION UNITS : TEMP BAND NUMBER : -1 INTERPOLATION TYPE: LIN
```

13. Display the second most recent GOES-16 2km IR image on frame 2 centered on Washington, DC, reduce the resolution by 2 and apply the stretch table MB. Draw a data bar on the image using a label interval of 10. Set the loop bounds from 1 to 2.
    Type: **IMGDISP G16FD.-1 2 STA=KDCA MAG=-2 BAND=13 SU=MB SF=YES REFRESH='BAR (GRA) LINT=10 COLOR=6';LS 1-2**
14. Compare the two images, one without a stretch table applied and the one with, as shown below.
    Press: **Alt B**
    Press: **Alt A**

Original image:



Data stretched image:



15. List the stretch tables on your workstation.
    Type: **SU LIST**

```
PERM SIZE LAST CHANGED FILENAME DIRECTORY ---- --------- ------------ ----------- --------- -rw- 1604 Oct 30 08:31
```

```
LEARN.ST /home/user/mcidas/data -rw- 1604 Oct 29 15:36 MB.ST /home/user/mcidas/data 3208 bytes in 2 files
```

16. Exit McIDAS
    Type: **EXIT**

MD Files

Table of Contents:

MD Files

SSEC receives real-time meteorological data that can be easily accessed and analyzed using McIDAS-X. Point data (i.e. observations at a particular location) are decoded and stored in McIDAS MD (Meteorological Data) files and/or database servers. The MD files contain information such as surface hourly weather data, radiosonde observations, ship and buoy observations, NCEP forecast guidance (FOUS), and surface synoptic data.

MD files can be used for creating plots, generating isentropic surfaces and stability parameters, and displaying soundings and cross sections. This lesson uses the basic commands to display, copy and list point data.

This lesson describes how to:

- list MD file directories and data records
- search MD files for specific observations
- copy and delete MD files
- display point data

The following commands are used in this lesson.

Command	Function
DMAP	lists information about files
DSINFO	lists ADDE datasets on local and remote servers
DSSERVE	manages dataset names on the local server
HODO	plots a hodograph of upper-air data
LSCHE	lists an MD file schema
MDU	MD file utility
PTCOPY	copies point data
PTDISP	displays point data
PTLIST	lists point data
RAOBCON	contours upper-air data from IRAB-schema point files
RAOBPLOT	plots upper-air data from IRAB-schema point files
SCHE	adds a schema to the schema file
SFCCON	contours surface hourly data from ISFC-schema point files
SFCLIST	lists surface hourly data from point files
SFCMG	displays a surface meteorogram
SFCPLOT	plots surface hourly data from ISFC-schema point files
UACROSS	plots a vertical cross section of upper-air data
UALIST	lists upper-air data
UAPLOT	plots a sounding on a thermodynamic diagram

MD Files - Basic Concepts

MD files normally contain observational data for a specific time period, for example, a day or an entire year. MD files store the data in individual records for a specific location at a specific time. Each MD file contains individual records. A record contains observational data for a latitude and longitude at a specific time. For example, one record may include measurements of temperature, dew point, wind speed, wind direction, and sea level pressure at 15 UTC for Houston, TX. A single MD file may contain thousands of records.

MD files use the naming convention MDXX*nnnn* where *nnnn* is a four-digit number. For example, MDXX0013 is the file name for MD file 13. Most McIDAS commands use only the MD file number. However, you must use the MDXX prefix with the DMAP command or when using Unix commands to copy, move, or delete MD files.

MD file Schemas

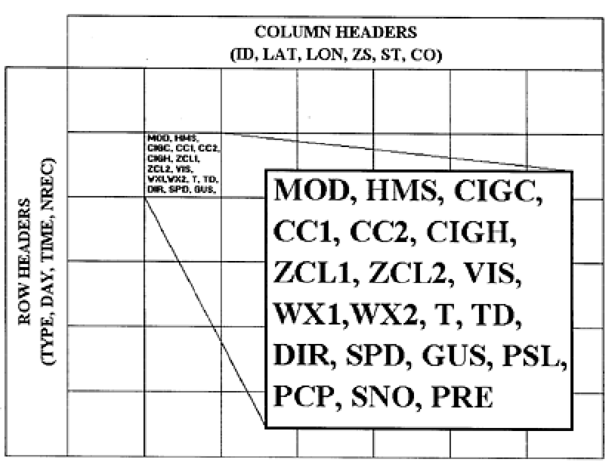
McIDAS stores MD files according to unique templates called *schemas*. SSEC receives real-time data for these nine schemas:

- ISFC (international surface hourly observations)
- IRAB (international RAOB (radiosonde) mandatory levels)
- IRSG (international RAOB (radiosonde) significant levels)
- ISHP (ship and buoy observations)
- PIRP (AIREP and PIREP reports)
- SYN (international surface synoptic observations)
- GFSMOS (real-time GFS MOS)
- NAMMOS (real-time NAM MOS)
- NGMMOS (real-time NGM MOS)

MD files in the ISFC, IRAB, SYN, and GFSMOS schemas are arranged in a table of rows and columns as shown in the ISFC example below. Time and day information common to all records in the row appears in the row header. Similarly, a column header designates common information according to location. Therefore, all the records along a particular row represent the same time and all the records down a particular column are reports from the same location.

MD files in the IRSG, ISHP, and PIRP schemas have row headers, but not column headers because the reporting location changes.

**Note: On 25 January 2005 the North American Mesoscale forecast model (NAM) replaced the ETA model, creating a new schema, NAMMOS. In addition, NGMMOS, which is the same as DCFO14, was created at the same time to follow the more logical naming convention of the other MOS schemas (GFSMOS and NAMMOS). References to older models (like NGM and ETA) are still found in the Learning Guide because it has many examples from the 1993 "Storm of the Century" and those models were operational at that time.



The individual data values within the MD file are stored according to the *parameters* of a schema. Examples of parameters within the ISFC schema are temperature, dew point, and cloud cover. Parameters are used for searching and plotting the MD file data. For each parameter, the schema provides:

- a scaling factor, for example 2, meaning temperatures are stored in Kelvin * 10**2
- the units in which the values are stored, for example Kelvin

SSEC's Real-time Point Data

The table below lists real-time point data that SSEC receives and the associated dataset names.

Schema	ADDE Dataset
PIRP	RTPTSRC/AIRCRAFT
GFSMOS	RTPTSRC/GFSMOS
NAMMOS	RTPTSRC/NAMMOS
NGMMOS	RTPTSRC/NGMMOS
ISFC	RTPTSRC/SFCHOURLY
ISHP	RTPTSRC/SHIPBUOY
SYN	RTPTSRC/SYNOPTIC
IRAB	RTPTSRC/UPPERMAND
IRSG	RTPTSRC/UPPERSIG

SSEC's History Data

SSEC has history data for the following schemas and years.

Schema	Years
ISFC	1976 to present
IRAB/IRSG	1977 to present
ISHP	1985 to present
SYN	1991 to present

MD Files - Listing MD File Directories and Data Records

In this exercise, you will list an MD file directory and information about its schema, and examine individual records in the MD file.

1. Start a McIDAS session.
   At the Unix prompt:
   Type: **mcidas**
   Your session should still be set for twenty frames from the last time you changed the Configuration GUI. If not, set it for twenty frames, as described in Getting Started.
2. Add a real time point server and list the point datasets available in the RTPTSRC dataset.
   Type: **DATALOC ADD RTPTSRC ADDE.SSEC.WISC.EDU**
   Type: **DSINFO POINT RTPTSRC**

```
DSINFO POINT RTPTSRC Dataset Names of Type: POINT in Group: RTPTSRC Name NumPos Content ------------ ------
------------------------------------- AIRCRAFT 10 Real-Time Aircraft data ETAMOS 10 Real-Time ETA Model Output
Statistics GFSMOS 10 Real-Time GFS Model Output Statistics PROF6MIN 10 Real-Time 6-Minute Profiler data PROFHOURLY
10 Real-Time Hourly Profiler data PTSRCS 130 All point data in MDXX files SFCHOURLY 10 Real-Time SFC Hourly SHIPBUOY
10 Real-Time Ship and Buoy data SYNOPTIC 10 Real-Time SYNOPTIC data TEMPSHIP 10 Real-Time TEMP SHIP data UPPERMAND
10 Real-Time Upper Air (Mandatory) UPPERSIG 10 Real-Time Upper Air (Significant) DSINFO -- done
```

3. List the file directories for the RTPTSRC/SFCHOURLY dataset.
   Type: **PTLIST RTPTSRC/SFCHOURLY FORM=FILE ALL**
   The dataset position number, schema type, number of rows and columns, and description for each MD file are listed.

```
PTLIST BLIZZARD/PTSRCS FORM=FILE ALL Pos Description Schema NRows NCols Proj# Created ------
------------------------------ ------ ----- ----- ----- ------- 1 SAO/METAR data for 30 APR 2020 ISFC 72 12500 0
2020120 2 SAO/METAR data for 01 MAY 2020 ISFC 72 12500 0 2020121 3 SAO/METAR data for 02 MAY 2020 ISFC 72 12500 0
2020122 4 SAO/METAR data for 03 MAY 2020 ISFC 72 12500 0 2020123 5 SAO/METAR data for 04 MAY 2020 ISFC 72 12500 0
2020124 6 SAO/METAR data for 05 MAY 2020 ISFC 72 12500 0 2020125 7 SAO/METAR data for 06 MAY 2020 ISFC 72 12500 0
2020126 10 SAO/METAR data for 29 APR 2020 ISFC 72 12500 0 2020119 PTLIST: Done
```

4. List all the MD files stored on your workstation.
   Type: **DMAP MDXX**
5. List the structure and keys in the ISFC schema.
   Type: **LSCHE ISFC**
   The keys, units, and scale factors for the ISFC schema are displayed as shown below. If the schema type does not exist, register the schema with the SCHE DCISFC command, and then try the LSCHE command again. See the introduction in the McIDAS User's Guide for more information.

```
LSCHE ISFC NAME: ISFC VERSION: 8 DATE: 2018073 TEXTID: "SURFACE HOURLY OBSERVATIONS ---- ------- DEFAULT NUMBER OF
ROWS: 240 INTEGER ID: 0 COLS: 12500 MISSING DATA VALUE: -2139062144 REPEAT GROUP: NUMBER OF REPETITIONS: 1 STARTING
POSITION: 11 SIZE: 25 NUMBER OF KEYS IN ROW HEADER: 4 COL HEADER: 6 STARTING AT POSITION 5 DATA RECORD: 25 STARTING
AT POSITION 11 ---- 35 TOTAL KEY SCALE UNIT KEY SCALE UNIT KEY SCALE UNIT ---- ----- ---- ---- ----- ---- ---- -----
---- TYPE 0 DAY 0 CYD TIME 0 HMS NREC 0 ID 0 CHAR LAT 4 DEG LON 4 DEG ZS 0 M ST 0 CHAR CO 0 CHAR MOD 0 HMS 0 HMS
CIGC 0 CC1 0 CC2 0 CIGH -2 FT ZCL1 -2 FT ZCL2 -2 FT VIS 1 MI WX1 0 CHAR WX2 0 CHAR T 2 K TD 2 K DIR 0 DEG SPD 1 MPS
GUS 1 MPS PSL 2 MB PCP 2 IN SNO 0 IN PRE 2 MB P24 2 IN WXC1 0 CODE WXC2 0 CODE WXC3 0 CODE WXC4 0 CODE LSCHE: DONE
```

6. List the parameters in the first file of the RTPTSRC/SFCHOURLY dataset.
   Type: **PTLIST RTPTSRC/SFCHOURLY FORM=PARAM**
   Also listed are the units, storage type and format for each parameter.

```
PTLIST RTPTSRC/SFCHOURLY FORM=PARAM PARAMETER UNIT STORAGE TYPE DEFAULT FORMAT --------- ---- ------------
-------------- TYPE integer I9 DAY CYD integer I9 TIME HMS integer I9 NREC integer I9 ID character A4 LAT DEG real
F9.4 LON DEG real F9.4 ZS M integer I9 ST character A4 CO character A4 MOD integer I9 HMS HMS integer I9 CIGC
integer I9 CC1 integer I9 CC2 integer I9 CIGH FT real F9 ZCL1 FT real F9 ZCL2 FT real F9 VIS MI real F9.1 WX1
character A4 WX2 character A4 T K real F9.2 TD K real F9.2 DIR DEG integer I9 SPD MPS real F9.1 GUS MPS real F9.1
PSL MB real F9.2 PCP IN real F9.2 SNO IN integer I9 PRE MB real F9.2 P24 IN real F9.2 WXC1 CODE integer I9 WXC2 CODE
integer I9 WXC3 CODE integer I9 WXC4 CODE integer I9 PTLIST: Done
```

7. List the first record in the most recent file of the dataset.
   Type: **PTLIST RTPTSRC/SFCHOURLY**

```
PTLIST RTPTSRC/SFCHOURLY Row : 1 Col : 1 TYPE = 0 | DAY = 2020127 CYD | TIME = 0 HMS | NREC = 8544 | ID = KEFK | LAT
= 44.8888 DEG | LON = 72.2125 DEG | ZS = 283 M | ST = VT | CO = US | MOD = 0 | HMS = 235600 HMS | CIGC = _missing_ |
CC1 = 1 | CC2 = _missing_ | CIGH = _missing_ | ZCL1 = 6000. FT | ZCL2 = _missing_ | VIS = 10.0 MI | WX1 = _missing_
| WX2 = _missing_ | T = 278.16 K | TD = 271.86 K | DIR = 0 DEG | SPD = 0.0 MPS | GUS = _missing_ | PSL = 1013.88 MB
| PCP = _missing_ | SNO = _missing_ | PRE = _missing_ | P24 = _missing_ | WXC1 = _missing_ | WXC2 = _missing_ | WXC3
= _missing_ | WXC4 = _missing_ | --------------------------------------------------------------------------- Number
of matches found = 1 PTLIST: Done
```

MD Files - Searching MD Files

The SFCLIST and UALIST commands list out weather observations in an easy to read tabular format. The default for these commands is to list the observations from the current day and time from the real-time datasets. By specifying the TIME, DAY, and DATASET keywords, you will can access specific data in other datasets. If you need to specify which parameters to list, or want to change the format of the output extensively, you will have to use the PTLIST command.

In this exercise, you will use the SFCLIST and UALIST commands to list out weather data for Wilmington, North Carolina, as well as for the entire state of North Carolina. You will then use the PTLIST command to format the data more specifically.

1. List 12 hours of surface hourly data for Wilmington, North Carolina for today. (DAY=#Y uses string replacement for the current day. Strings will be discussed in String Tables.) The default dataset is used, RTPTSRC/SFCHOURLY.
   Type: **SFCLIST KILM TIME=0 12 DAY=#Y**
2. List the surface hourly data for North Carolina at 18 UTC, including the windchill, heat index and precipitation information.
   Type: **SFCLIST NC TIME=18 DAY=#Y OPT=CHILL HEAT PRECIP**
3. List the upper-air data from 00 UTC at Newport, North Carolina.
   Type: **UALIST KMHX 00 #Y**
4. List the station identifier (ID), temperature (T) and dew point (TD) of the matching observations for North Carolina at 15 UTC today. Display the temperature and dew point in Fahrenheit. The PARAM keyword defines what data types in the matching records to list and the SELECT keyword defines which records to match. You must surround the SELECT variable with single quotes and separate individual select clauses with semicolons.
   Type: **PTLIST RTPTSRC/SFCHOURLY PARAM=ID T[F] TD[F] SELECT='ST NC;TIME 15' FORMAT=X I6 I6 NUM=ALL**
5. List all records between 9 UTC and 12 UTC for today in Wilmington, North Carolina.
   Type: **PTLIST RTPTSRC/SFCHOURLY SELECT='ID KILM;TIME 9 12' NUM=ALL**
   The ALL parameter lists all reporting stations in the MD file that match the criteria. Without the ALL parameter only the first match is listed. Also, without the PARAM= keyword, the raw records are listed.

MD Files - Copying and Deleting MD Files

In this exercise, you will copy MD file data from one file to another on the workstation and delete an MD file.

1. Use the DMAP command to see if there are any local MD files in the range 4000 to 4005.
   Type: **DMAP MDXX400**
   If you already have MD files in this range, select another range of 6 files you can use. (try 4010-4015, then 4020-4025, etc).

2. Create a local dataset name for the test MD files in the MYDATA group.
   Type: **DSSERVE ADD MYDATA/TEST-PTSRCS MD 4000 4005**
3. Copy the contents of RTPTSRC/SFCHOURLY.2 to MYDATA/TEST-PTSRCS.1 (MD file 4000), and change the file description to CASE STUDY.
   Type: **PTCOPY RTPTSRC/SFCHOURLY.2 MYDATA/TEST-PTSRCS.1 DEL=YES TITLE='CASE STUDY'**
4. Once the PTCOPY is done, list the MD file headers to verify that the MD file was copied.
   Type: **PTLIST RTPTSRC/SFCHOURLY.2 FORM=FILE**
   Type: **PTLIST MYDATA/TEST-PTSRCS.1 FORM=FILE**
5. In step 3, you copied all of the data available in RTPTSRC/SFCHOURLY.2 to your local dataset. You may also choose which data to copy by using the SELECT clause. Copy just today's 12Z mandatory upper air data from selected states to position 2 of your local dataset (MD file 4001).
   Type: **PTCOPY RTPTSRC/UPPERMAND MYDATA/TEST-PTSRCS.2 DEL=YES SEL='DAY #Y;TIME 12;ST MD,VA,NC,SC,GA'**
6. List the data that you just copied.
   Type: **PTLIST MYDATA/TEST-PTSRCS.2 PARAM=DAY TIME ST IDN NUM=ALL**
   Only data from the five states listed in step 6 should be included in the output.

7. Since you won't be using MD file 4000 any longer, delete it.
   Type: **MDU DEL 4000**

MD Files - Plotting and Contouring Point Data

The SFCCON, SFCPLOT, RAOBCON, and RAOBPLOT commands display surface and upper-air point data. These commands are designed to allow for shorter command entries by having commonly used defaults for displaying their data types, and using positional parameters instead of keywords for values that are always or often specified.

For example, the default for the DATASET keyword is to use the real-time datasets. When looking at real-time data you will never have to specify the DATASET keyword. when using McIDAS-XCD default named datasets.

These commands are easier to use for plotting or contouring their specific data types than the generic "toolbox" commands PTDISP and PTCON. However, the "toolbox" commands are still useful for calculating complicated mathematical equations and for creating output that is completely controlled by the user instead of by the command defaults.

In this exercise, you will plot and contour observed parameters, as well as calculate the changes in these parameters over time and between different levels in the atmosphere. You will also plot some derived parameters and plot the results of your own mathematical equation.

Displaying Surface Data

1. Erase both the images and graphics in frames 1 through 6.
   Type: **ERASE F 1 6**
2. Display two current GOES-16 band 2 Vis images on frames 1 and 2 centered on Florence, South Carolina, with the resolution reduced by a factor of 2. Show each frame and add a map in graphics color level 5.
   Type: **IMGDISP RTGOESR/CONUSC02.-2 1 STA=KFLO MAG=-2 SF=YES REFRESH='MAP H 5'**
   Type: **IMGDISP RTGOESR/CONUSC02.-1 2 STA=KFLO MAG=-2 SF=YES REFRESH='MAP H 5'**
3. Show frame 1 and plot the temperature on the satellite image using data from the RTPTSRC/SFCHOURLY dataset. The **X** in the *map* positional parameter means to use the default, which is to use the frame's current navigation.
   Type: **SF 1;SFCPLOT T X**
4. Contour the temperature over the plot using the same data and conditions as in the previous step.
   Type: **SFCCON T X**
5. On frame 2, contour the pressure change from 1 UTC to 2 UTC.
   Type: **SFCCON PRE X 1-2 GRA=2;SF 2**

Displaying Upper-air Data

1. Display a map of the United States in color level 8 (gray) on frames 3 and 4.
   Type: **SF 3;MAP USA 8 GRA=3-4**
2. Contour the 850-500 mb thickness on frame 3.
   Type: **RAOBCON Z 500-850 X 12 #Y**
3. Plot and contour the 850 mb temperature over the United States at 12 UTC on frame 4.
   Type: **SF 4;RAOBPLOT T 850 X 12 #Y**
   Type: **RAOBCON T 850 X 12 #Y**
4. Plot the 850 mb streamlines over the temperature display in red.
   Type: **RAOBCON STREAML 850 X 12 #Y COL=5**

Displaying Calculated Parameters

In addition to plotting the observed parameters, these commands can also calculate and plot some derived parameters, such as windchill, vorticity, and temperature advection. If you have a more complicated equation that you would like to use, you can use the PTCON or PTDISP commands to plot the results of your equation.

1. Contour the 850 mb temperature advection on frame 5, using a contour interval of 5.
   Type: **SF 5;RAOBCON TADV 850 USA 12 #Y CINT=5**
2. Compare frames 4 and 5.
   Press: **Alt B**
   Press: **Alt A**
   Notice that the areas of positive temperature advection (warm air advection) calculated and plotted in frame 5 match the areas in frame 4 where streamlines are passing from warmer to colder air.

3. Use the Hypsometric Equation to calculate the 1000-500 mb thickness in meters.
   Type: **PTDISP RTPTSRC/UPPERMAND 6 SEL='DAY #Y;TIME 12' MAP=USA PARAM='THIK[M]=(287/9.8)*((P1+P2)/2)*LOG(1000/500)' P1=T[K] 'P 1000' P2=T[K] 'P 500' ; SF 6**
   Only stations reporting both a 1000 mb and 500 mb temperature are included in the equation, so stations that are above 1000 mb (e.g. Rocky Mountain stations) will not appear in the display.

MD Files - Meteorological Diagrams

In this exercise, you will create four types of meteorological diagrams: a Skew-T diagram, a hodograph, a vertical cross section and a surface meteorogram. You will also list surface data in a tabular format.

1. Erase the images and graphics in frames 1 through 6.
   Type: **ERASE F 1 6**
2. List the upper-air reporting stations in position 2 of the MYDATA/TEST-PTSRCS dataset that you copied in the previous exercise.
   Type: **PTLIST MYDATA/TEST-PTSRCS.2 PAR=IDN TIME DAY NUM=ALL**
3. Choose one of the stations listed. Show frame 1 and plot a Skew-T diagram for your chosen station (in this case, 72208, which is Charleston, South Carolina) at 12 UTC using the data in MYDATA/TEST-PTSRCS.2. Note that the actual day needs to be specified if #Y doesn't match.
   Type: **SF 1;UAPLOT 72208 12 #Y DATASET=MYDATA/TEST-PTSRCS.2**
4. Plot a hodograph of the data from Charleston on frame 2. Show frame 2
   Type: **HODO 72208 12 #Y GRA=2 DATASET=MYDATA/TEST-PTSRCS.2;SF 2**
5. Show frame 3 and display a map of the stations contained in this dataset.
   Type: **SF 3;PTDISP MYDATA/TEST-PTSRCS.2 PARAM=IDN MAP=USA**
6. Choose two stations that are the farthest apart on the map. Show frame 4 and display a vertical cross section of theta, wind, and mixing ratio values from these two stations (in this case, Peachtree City, Georgia (72215) to Wallops, Virginia (72402)) using upper air data from MD file 4001. The MAP=YES keyword displays a map in the upper-right corner showing the cross section location and stations selected.
   Type: **UACROSS 72215 72402 MAP=YES DATASET=MYDATA/TEST-PTSRCS.2 TIME=12 DAY=#Y PARAM=THA WINDB MIX GRA=4;SF 4**
7. Show frame 5 and plot a 24-hour surface meteorogram for Middle Georgia Airport, ending at 23 UTC on the previous day.
   Type: **SF 5;SFCMG KMCN TIME=23 DAY={enter yesterday's Julian date}**
8. List the surface hourly data for Middle Georgia Airport for the same 24-hour period.
   Type: **SFCLIST KMCN OPT=PRECIP TIME=0 23 DAY={enter yesterday's Julian date}**
   You should notice that the data listed in the Text Window is the same as that displayed in the Image Window.

9. Exit McIDAS.
   Type: **EXIT**

Grids and Grid Files

Table of Contents:

Grids and Grid Files

A McIDAS grid is a lattice of regularly spaced data points. Grids can be generated from numerical models or observational data. When you contour observational data, it is interpolated and can be stored as a grid. Grids are stored in grid files.

This lesson describes how to:

- list grids and grid files
- copy grids and grid files
- create grids and grid files
- display data from grid files

The following commands are used in this lesson:

Command	Function
GRDCOPY	copies grids from one dataset to another
GRDDISP	generates contours, streamlines or plots from grid data
GRDINFO	lists, modifies, or statistically analyzes grid point values
GRDLIST	lists grids in a dataset

Grids and Grid Files - Basic Concepts

Grids

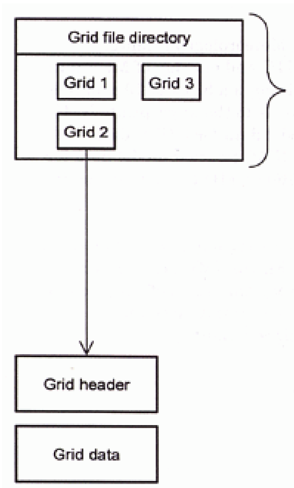
A grid is a lattice of regularly spaced data points superimposed on a projection of the earth. Grids can store numerical model data and serve as an interface between observational data stored in MD files and analyzed data displayed as contours on a frame. To contour observational data, compute derived parameters, and display manually digitized radar (MDR) data, etc., the data must first be interpolated from the observing locations to a uniform latitude, longitude lattice. McIDAS stores this geographic lattice as a grid.

Grid Files

Grids are stored in grid files, GRIB files or netCDF files. This manual will use a local grib file and real time grib/grid servers in the real time RTGRIDS dataset. McIDAS Grid files use the naming convention GR*nnnnnn* where *nnnnnn* is the six-digit file number. If the file number is less than six digits, I and D replace the first two digits and zeros precede the number. For example, GR111111 is the name of the file for grid 111111, GRI90112 is the name for grid file 90112, and GRID0013 is the name for grid file 13. In ADDE, sequences of grid files are grouped together in a dataset.

Grid File Format

Each grid file contains a grid file directory and up to 10,000 grids. Each grid contains a grid header and grid data, as shown below. A partial grid dataset directory and a listing of grid header information are also shown below.




```
GRDLIST RTGRIDS/GFS FORM=FILE DATASET NAME: RTGRIDS/GFS Dataset Position Creation Date Max
Grids Directory Title ---------------- ------------- --------- ------------------------------
3 2020133 20000 ALL 00Z GFS 0 HR<=VT<=24 HR 4 2020134 20000 ALL 00Z GFS 0 HR<=VT<=24 HR 5
2020135 20000 ALL 00Z GFS 0 HR<=VT<=24 HR 13 2020133 20000 ALL 00Z GFS 24 HR<=VT<=48 HR 14
2020134 20000 ALL 00Z GFS 24 HR<=VT<=48 HR
```

```
GRDLIST RTGRIDS/GFS FORM=ALL Dataset position 3 Directory Title= ALL 00Z GFS 0 HR<=VT<=24 HR
PAR LEVEL DAY TIME SRC FHR FDAY FTIME GRID PRO ---- ---------- ------------ -------- ---- ----
------------ -------- ----- ---- Z 550 MB 12 MAY 20133 00:00:00 GFS 6 12 MAY 20133 06:00:00 1
LAMB Total pts= 6045 Num rows= 65 Num columns= 93 received: 2020133 033333Z Geopotential
height GRIB ID numbers: Geographic = 211; PAR = 7; Model ID = 96; Level type =100 Units of
gridded variable are GPM Scale of variable is: 2 Lambert Conformal Tangent Cone Projection Row
num of pole= -113.34 Col num of pole= 53.00 Col spacing (m)= 81270.0 Standard Latitudes= 25.00
25.00 Standard Longitude= 95.00 Number of grids listed = 1 GRDLIST - done
```

Manipulating Grids and Grid Files

The GRDLIST and GRDCOPY commands manipulate individual grids or groups of grids. GRDLIST can list grids, while GRDCOPY can copy, move, and perform mathematical operations on grids. These operations include adding or subtracting two grids, filling a grid with a constant value, or redefining all values within a grid that are less than a specified number to be a new number (e.g., all values less than zero are set to zero). If you have u and v component grids, you can create a vorticity or divergence grid, or use the grid components to advect a parameter in another grid.

The GRDINFO command provides information about grids. You can use it to list information about the grid header and statistics on the grid data, or list data points.

The GRDLIST and GRDCOPY commands can also manipulate grid files. With these commands you can create, list, or copy a grid file.

SSEC's Real-time Grid Data

SSEC receives real-time grid data for many numerical models via the NOAAPORT data stream. The models and datasets change frequently so the following list is just a subset of grid dataset names and models that are available.

ADDE Dataset	Description
RTGRIDS/ALL	All model grids decoded by McIDAS-XCD
RTGRIDS/AWC-ALL	National Convective Weather Diagnostic grids
RTGRIDS/AWR-ALL	Alaska Waters Regional Wave Model grids
RTGRIDS/ENP-ALL	Eastern North Pacific Regional Wave Model grids
RTGRIDS/FFG-ALL	NWS Flash Flood Guidance System grids

RTGRIDS/GFS-ALL	Global Forecast System grids
RTGRIDS/ICA-ALL	Ice Concentration Analysis grids
RTGRIDS/ICN-ALL	Current Icing Potential grids
RTGRIDS/MDR-ALL	Manually Digitized Radar grids
RTGRIDS/NAM-ALL	North American Mesoscale Model grids
RTGRIDS/NCE-ALL	NCEP quantitative precipitation forecast grids
RTGRIDS/RAP-ALL	Rapid Refresh model grids
RTGRIDS/SST-ALL	Sea Surface Temp Analysis grids
RTGRIDS/UKM-ALL	Global Coastal Ocean Circulation grids
RTGRIDS/WHG-ALL	Great Lakes wind wave height grids
RTGRIDS/WNA-ALL	Western North Atlantic Regional Wave Model grids
RTGRIDS/WWF-ALL	Global Wind-Wave Forecast grids

Grids and Grid Files - Listing Grids and Grid Files

In this exercise, you will list the grid files in the real time RTGRIDS dataset and and a local grib file, list the grids in one of the grid files, and list information about the values in a grid.

1. Start a McIDAS session.
   At the Unix prompt:
   Type: `mcidas`
   Your session should still be set for twenty frames from the last time you changed the Configuration GUI. If not, set it for twenty frames, as described in Getting Started.
2. Add the RTGRIDS dataset and list the grid datasets in the RTGRIDS group.
   Type: **DATALOC ADD RTGRIDS ADDE.SSEC.WISC.EDU;DSINFO GRID RTGRIDS**
3. List the grid file directory for the GFS data.
   Type: **GRDLIST RTGRIDS/GFS.ALL FORM=FILE**
   Each grid file has a dataset position, date the grid file was created, maximum number of grids the grid file can store, and a file description, as shown below in the partial listing.

```
DATASET NAME: RTGRIDS/GFS Dataset Position Creation Date Max Grids Directory Title ---------------- -------------
--------- ------------------------------ 9 2020139 20000 ALL 00Z GFS 0 HR<=VT<=24 HR 10 2020140 20000 ALL 00Z GFS 0
HR<=VT<=24 HR 19 2020139 20000 ALL 00Z GFS 24 HR< VT<=48 HR 20 2020140 20000 ALL 00Z GFS 24 HR< VT<=48 HR 29 2020139
20000 ALL 00Z GFS 48 HR< VT<=72 HR 30 2020140 20000 ALL 00Z GFS 48 HR< VT<=72 HR
```

4. Setup a local dataset that accesses a local grib file and list the first ten grids in the first grid file in the dataset. Download the grib files from a FTP site. For this example, it is assumed the data has been placed in the directory ***<local-path>/Data/Grid*** directory.
   Download the files from ftp.ssec.wisc.edu.
   ftp ftp.ssec.wisc.edu
   login anonymous
   prompt
   binary
   cd pub/mug/Data/Grid
   mget GFS*
   Example file name:
   GFS_CONUS_80km_20190903_1200.grib1
5. Run the DSSERVE command to access the grib file.
   Type: **DSSERVE ADD GRIB/GFS GRIB TYPE=GRID DIRFILE='<local-path>/Data/Grid/GFS_CONUS*'**
6. List the first ten grids in the first grid file in the dataset.
   Type: **GRDLIST GRIB/GFS.1 NUM=10**
   The first ten grids are listed in the Text and Command Window, which shows the dataset position, grid file description, parameter type, level, Julian date of the data, time of the data, source type, forecast hour, valid day/time, grid number, and projection.

```
Dataset position 1 Directory Title= /GFS_CONUS_80km_20190903_1200.gr PAR LEVEL DAY TIME SRC FHR FDAY FTIME GRID PRO
---- ---------- ------------ -------- ---- ---- ------------ -------- ----- ---- Z 850 MB 03 SEP 19246 12:00:00 GFS
0 03 SEP 19246 12:00:00 1 LAMB Z 1000 MB 03 SEP 19246 12:00:00 GFS 0 03 SEP 19246 12:00:00 2 LAMB Z 750 MB 03 SEP
19246 12:00:00 GFS 0 03 SEP 19246 12:00:00 3 LAMB Z 900 MB 03 SEP 19246 12:00:00 GFS 0 03 SEP 19246 12:00:00 4 LAMB
Z 700 MB 03 SEP 19246 12:00:00 GFS 0 03 SEP 19246 12:00:00 5 LAMB Z 875 MB 03 SEP 19246 12:00:00 GFS 0 03 SEP 19246
12:00:00 6 LAMB Z 625 MB 03 SEP 19246 12:00:00 GFS 0 03 SEP 19246 12:00:00 7 LAMB Z 950 MB 03 SEP 19246 12:00:00 GFS
0 03 SEP 19246 12:00:00 8 LAMB Z 925 MB 03 SEP 19246 12:00:00 GFS 0 03 SEP 19246 12:00:00 9 LAMB Z 550 MB 03 SEP
19246 12:00:00 GFS 0 03 SEP 19246 12:00:00 10 LAMB Number of grids listed = 10
```

7. Get the expanded listing of the first grid in this dataset.
   Type: **GRDLIST GRIB/GFS.1 FORM=ALL**
   The expanded listing includes information about the total points, the number or rows and columns, the time the grid was received, units of the parameter, scaling factor, lat/lon extents and increments.

```
Dataset position 1 Directory Title= /GFS_CONUS_80km_20190903_1200.gr PAR LEVEL DAY TIME SRC FHR FDAY FTIME GRID PRO
---- ---------- ------------ -------- ---- ---- ------------ -------- ----- ---- Z 850 MB 03 SEP 19246 12:00:00 GFS
0 03 SEP 19246 12:00:00 1 LAMB Total pts= 6045 Num rows= 65 Num columns= 93 received: 2020141 013638Z Geopotential
height GRIB ID numbers: Geographic = 211; PAR = 7; Model ID = 81; Level type =100 Units of gridded variable are GPM
Scale of variable is: 2 Lambert Conformal Tangent Cone Projection Row num of pole= -113.34 Col num of pole= 53.00
Col spacing (m)= 81270.0 Standard Latitudes= 25.00 25.00 Standard Longitude= 95.00 Number of grids listed = 1
```

8. List all the 500 mb height (Z) grids in the grib file.
   Type: **GRDLIST GRIB/GFS LEVEL=500 PARAM=Z NUM=ALL**
   The forecast grids for this level and parameter are listed.

9. List the statistical information for the 12 hour 500 mb height grid in this dataset.
   Type: **GRDINFO GRIB/GFS STAT LEVEL=500 PARAM=Z FHOUR=12**
   Information about the maximum and minimum values, the mean and standard deviation of the values in the grid are listed.

10. List a histogram of the values for this same grid. Bin the values in 120 m segments and format the values as integers.
    Type: **GRDINFO GRIB/GFS HIST LEVEL=500 PARAM=Z FHOUR=12 BINSIZE=120 FORMAT=I5**
11. List the 24 hour Relative Humidity grid point values between 35° and 40° N and 80° and 85° W from this dataset. Format the data as a real value with one decimal place.
    Type: **GRDINFO GRIB/GFS LIST PARAM=RH FHOUR=24 LAT=35 40 LON=80 85 FORMAT=F4.1**
    Row, column, latitude, longitude, and values are listed for each grid point falling in this geographical region:

```
Listing data from 24-hour forecast RH Grid at 925 MB from GFS Grid is at 120000 UTC on 2019246 from dataset:
```

```
GRIB/GFS Row Col Latitude Longitude Value Units Level Param 34 64 39.721 84.897 76.3 % 925 MB RH 35 64 39.016 84.965
91.2 % 925 MB RH 34 65 39.666 83.982 85.7 % 925 MB RH 35 65 38.961 84.056 97.8 % 925 MB RH 36 65 38.254 84.129 76.2
% 925 MB RH 37 65 37.544 84.201 59.9 % 925 MB RH 38 65 36.833 84.272 56.5 % 925 MB RH 39 65 36.119 84.342 58.3 % 925
MB RH 40 65 35.404 84.412 55.2 % 925 MB RH 34 66 39.606 83.069 95.2 % 925 MB RH 35 66 38.902 83.149 87.0 % 925 MB RH
36 66 38.195 83.228 64.7 % 925 MB RH 37 66 37.485 83.306 58.8 % 925 MB RH 38 66 36.774 83.383 57.1 % 925 MB RH 39 66
36.061 83.458 55.2 % 925 MB RH 40 66 35.346 83.533 86.2 % 925 MB RH 34 67 39.542 82.156 96.1 % 925 MB RH 35 67
38.837 82.242 79.4 % 925 MB RH 36 67 38.131 82.327 66.1 % 925 MB RH 37 67 37.422 82.411 55.7 % 925 MB RH 38 67
36.711 82.494 63.6 % 925 MB RH 39 67 35.998 82.575 77.9 % 925 MB RH 40 67 35.283 82.656 79.9 % 925 MB RH 34 68
39.473 81.245 91.7 % 925 MB RH 35 68 38.768 81.337 72.2 % 925 MB RH 36 68 38.062 81.428 58.7 % 925 MB RH 37 68
37.353 81.518 62.6 % 925 MB RH 38 68 36.642 81.606 77.9 % 925 MB RH 39 68 35.930 81.693 72.4 % 925 MB RH 40 68
35.215 81.780 76.5 % 925 MB RH 34 69 39.399 80.335 77.9 % 925 MB RH 35 69 38.695 80.433 64.7 % 925 MB RH 36 69
37.988 80.530 72.1 % 925 MB RH 37 69 37.280 80.625 71.9 % 925 MB RH 38 69 36.569 80.719 68.0 % 925 MB RH 39 69
35.857 80.813 76.7 % 925 MB RH 40 69 35.143 80.904 72.3 % 925 MB RH 40 70 35.066 80.030 72.1 % 925 MB RH GRDINFO
Done, Number of grids listed=1
```

Grids and Grid Files - Copying Grids and Grid Files

Next, you will copy grids and grid files.

1. Use the DMAP command to see if there are any local grid files in the range 4000 to 4009.
   Type: **DMAP GRID400**
   If you already have grids in this range, select another range of 10 files you can use. (try 4010-4019, then 4020-4029, etc).

2. Create a local dataset name for the test grids in the MYDATA group and add an alias for the dataset. Use the grid numbers for the first free range you located in the previous step.
   Type: **DSSERVE ADD MYDATA/TEST-GRIDS GRID 4000 4009 "Test grids**
   Type: **AKA ADD TG MYDATA/TEST-GRIDS**
3. Copy 20 parameters of the 0Z current day FHOUR=24 model run from the RTGRIDS/GFS dataset to the first position in the TEST-GRIDS dataset. Set the description of the destination grid file to "LEARNING HOW TO COPY GRIDS".
   Type: **GRDCOPY RTGRIDS/GFS TG.1 DEL=YES DAY=#Y TIME=0 FHOUR=24 NUM=20 TITLE='LEARNING HOW TO COPY GRIDS'**
4. List all of the copied grids, then list grids 5 through 10 in dataset TG.1.
   Type: **GRDLIST TG.1 NUM=ALL**
   Type: **GRDLIST TG.1 GRID=5 10**
   Note that grids 1 through 20 are used; therefore the first available grid is 21 in this dataset.

5. Copy 5 grids with FHOUR=12 from the same day to the TG.1 dataset starting at grid 21.
   Type: **GRDCOPY RTGRIDS/GFS TG.1 DAY=#Y TIME=0 FHOUR=12 NUM=5 DGRID=21**
6. Verify the grids were copied into the TG.1 dataset; grids 1 through 25 are now being used. (If less than 20 grids were copied in step 3, there might be a gap in the grid numbers in TG.1.)
   Type: **GRDLIST TG.1 GRID=1 25**
   Notice that grids 1 to 20 have the FHOUR of 24, and grids 21 to 25 have the FHOUR of 12.

```
Dataset position 1 Directory Title= LEARNING TO COPY GRIDS PAR LEVEL DAY TIME SRC FHR FDAY FTIME GRID PRO ----
---------- ------------ -------- ---- ---- ------------ -------- ----- ---- V 650 MB 22 MAY 20143 00:00:00 GFS 24 23
MAY 20144 00:00:00 1 LAMB U 250 MB 22 MAY 20143 00:00:00 GFS 24 23 MAY 20144 00:00:00 2 LAMB Z 200 MB 22 MAY 20143
00:00:00 GFS 24 23 MAY 20144 00:00:00 3 LAMB V SFC MBAG 22 MAY 20143 00:00:00 GFS 24 23 MAY 20144 00:00:00 4 LAMB U
60 MBAG 22 MAY 20143 00:00:00 GFS 24 23 MAY 20144 00:00:00 5 LAMB V 500 MB 22 MAY 20143 00:00:00 GFS 24 23 MAY 20144
00:00:00 6 LAMB U 825 MB 22 MAY 20143 00:00:00 GFS 24 23 MAY 20144 00:00:00 7 LAMB T TRO 22 MAY 20143 00:00:00 GFS
24 23 MAY 20144 00:00:00 8 LAMB V 120 MBAG 22 MAY 20143 00:00:00 GFS 24 23 MAY 20144 00:00:00 9 LAMB V 1000 MB 22
MAY 20143 00:00:00 GFS 24 23 MAY 20144 00:00:00 10 LAMB U 725 MB 22 MAY 20143 00:00:00 GFS 24 23 MAY 20144 00:00:00
11 LAMB WP 500 MB 22 MAY 20143 00:00:00 GFS 24 23 MAY 20144 00:00:00 12 LAMB V 700 MB 22 MAY 20143 00:00:00 GFS 24
23 MAY 20144 00:00:00 13 LAMB Z 550 MB 22 MAY 20143 00:00:00 GFS 24 23 MAY 20144 00:00:00 14 LAMB WP 850 MB 22 MAY
20143 00:00:00 GFS 24 23 MAY 20144 00:00:00 15 LAMB U SFC MBAG 22 MAY 20143 00:00:00 GFS 24 23 MAY 20144 00:00:00 16
LAMB U 150 MB 22 MAY 20143 00:00:00 GFS 24 23 MAY 20144 00:00:00 17 LAMB WP 200 MB 22 MAY 20143 00:00:00 GFS 24 23
MAY 20144 00:00:00 18 LAMB T 90 MBAG 22 MAY 20143 00:00:00 GFS 24 23 MAY 20144 00:00:00 19 LAMB LI SFC 22 MAY 20143
00:00:00 GFS 24 23 MAY 20144 00:00:00 20 LAMB U SFC MBAG 22 MAY 20143 00:00:00 GFS 12 22 MAY 20143 12:00:00 21 LAMB
U TRO 22 MAY 20143 00:00:00 GFS 12 22 MAY 20143 12:00:00 22 LAMB U 525 MB 22 MAY 20143 00:00:00 GFS 12 22 MAY 20143
12:00:00 23 LAMB U 120 MBAG 22 MAY 20143 00:00:00 GFS 12 22 MAY 20143 12:00:00 24 LAMB U 725 MB 22 MAY 20143
00:00:00 GFS 12 22 MAY 20143 12:00:00 25 LAMB Number of grids listed = 25 GRDLIST - done
```

Grids and Grid Files - Creating Grids and Grid Files

In this exercise, you will create a new grid dataset, copy and contour grids, create advection and divergence grids, and list the grids and grid files.

1. Generate a local grid dataset in position 2 of the TEST-GRID dataset with a maximum of 100 grids, and the description ADVECTION AND DIVERGENCE GRIDS. Copy in the 12 hour 1000 mb forecast grids from the GRIB/GFS dataset.
   Type: **GRDCOPY GRIB/GFS TG.2 DEL=YES LEVEL=1000 FHOUR=12 TITLE='ADVECTION AND DIVERGENCE GRIDS' MAXGRD=100 NUM=ALL**
   If you don't specify a maximum number of grids, the grid file is created with space for 1000 grids.

2. List the new dataset.
   Type: **GRDLIST TG.2 NUM=ALL**
   The dataset should contain five grids.

3. Check to see that TG.2 contains the grids required to calculate divergence.
   Type: **GRDLIST TG.2 DER=DVG**
   If the grid can be created, the following output will be shown:
   ```
   GRDLIST TG.2 DER=DVG PAR LEVEL DAY TIME SRC FHR FDAY FTIME GRID PRO ---- ---------- ------------ -------- ---- ---- ------------ -------- ----- ---- DVG 1000 MB 03 SEP 19246 12:00:00 GFS 12 04 SEP 19247 00:00:00 N/A LAMB Number of grids listed = 1 GRDLIST - done
   ```

   Notice the GRID number is listed as "MATH" or "N/A". This indicates that there is no actual divergence grid but that the component grids required to create it all exist within the TG.2 grid file. See the help sections of the individual commands for more information on the common meteorological parameters you can calculate from grids by using the DERIVE keyword.

   If the grid cannot be created because the component grids do not exist, an error will be produced.

4. Create a divergence grid from from the grids in the dataset.
   Type: **GRDCOPY TG.2 TG.2 DER=DVG**
5. List the contents of the dataset to verify that a divergence grid was added.
   Type: **GRDLIST TG.2 NUM=ALL**
   The new grid was filed into grid 6, which is the first available grid:
   ```
   Dataset position 2 Directory Title= ADVECTION AND DIVERGENCE GRIDS PAR LEVEL DAY TIME SRC FHR FDAY FTIME GRID PRO ---- ---------- ------------ -------- ---- ---- ------------ -------- ----- ---- U 1000 MB 03 SEP 19246 12:00:00 GFS 12 04 SEP 19247 00:00:00 1 LAMB Z 1000 MB 03 SEP 19246 12:00:00 GFS 12 04 SEP 19247 00:00:00 2 LAMB V 1000 MB 03 SEP 19246 12:00:00 GFS 12 04 SEP 19247 00:00:00 3 LAMB RH 1000 MB 03 SEP 19246 12:00:00 GFS 12 04 SEP 19247 00:00:00 4 LAMB T 1000 MB 03 SEP 19246 12:00:00 GFS 12 04 SEP 19247 00:00:00 5 LAMB DVG 1000 MB 03 SEP 19246 12:00:00 GFS 12 04 SEP 19247 00:00:00 6 LAMB Number of grids listed = 6
   ```

   Notice that the the DVG grid has now been created, has a grid number, and is listed as a PARAMETER.

6. List information about the divergence grid:
   Type: **GRDINFO TG.2 PARAM=DVG**
   ```
   GRDINFO TG.2 PARAM=DVG Statistics of Parameter: DVG Units: 1/S Level of data: 1000 MB Initial time of data: 120000 UTC Initial day of data: 2019246 Data came from: GFS 12-hour forecast Dataset: TG Minimum: -0.2449790E-03 occurred at [row,col]: [ 49, 72] [lat,lon]: [ 28.42, 79.20] Maximum: 0.1370970E-03 occurred at [row,col]: [ 39, 37] [lat,lon]: [ 35.86, 109.19] Mean: -0.2564339E-06 SD : 0.2353742E-04 Number of points analyzed: 6045 Number of points missing: 0 Row range: 1 to 65 Col range: 1 to 93 GRDINFO Done, Number of grids statistically analyzed=1
   ```
7. Create a temperature advection grid using the U, V and T grids. Use the formula TADV = -(u*DDX(T) + v*DDY(T)). Label the new grid TADV and set the units to K/sec.
   Type: **GRDCOPY TG.2 TG.2 G1='PARAM T' G2='PARAM U' G3='PARAM V' MATH='-(G2*(DDX(G1))+G3*(DDY(G1)))' NEWPAR=TADV K/S**
   The G1, G2, ..., Gn keywords are used to define the grids you want to use. Since all the grids in this dataset are from the same level, model run and forecast hour, just specifying the parameter defines the grid uniquely. You can use additional search criteria (e.g. LEVEL, FHOUR) to further refine your selection. The MATH keyword defines the mathematical operation to perform on the grids specified with the *Gn* keywords. See the help sections of the individual commands for more information on the use of these keywords.

8. List information about the temperature advection grid:
   Type: **GRDINFO TG.2 PARAM=TADV**
9. OR, if you would like to get statistical information on a grid created with a mathematical equation, but won't need to actually display the grid, you can run the GRDINFO command with the MATH keywords included.
   Type: **GRDINFO TG.2 G1='PARAM T' G2='PARAM U' G3='PARAM V' MATH='-(G2*(DDX(G1))+G3*(DDY(G1)))' NEWPAR=TTAD K/S**
   This should list the same information as the previous GRDINFO command, except with a different parameter name.

In the next lesson, you will display the grids you created using the GRDDISP command.

Grids and Grid Files - Displaying Gridded Data

In this exercise, you will use the GRDDISP command to display grids. GRDDISP can be used to draw contours, streamlines, or plots of grid data.

1. Show frame 4, and display the first GOES-16 0.5km Visible image centered on Washington DC. Add a high resolution map.
   Type: **SF 4;IMGDISP ABI/CONUS.1 4 STA=DCA BAND=2 MAG=-8;MAP H**
2. Contour the divergence grid created in the previous lesson over the satellite image with a contour interval of 5 and graphics color level 3. Because the values of divergence are small, use the POWER keyword to scale the values. Make the convergence (negative divergence) areas dashed. Use NAV=C to use the navigation in the current frame.
   Type: **GRDDISP TG.2 PARAM=DVG NAV=C CINT=5 COLOR=3 POWER=6 DASH=NEG**
3. Contour the temperature advection grid over the image using a contour interval of five degrees per sec scaled by 10**5. Make negative values (cold air advection) dashed.
   Type: **GRDDISP TG.2 PARAM=TADV NAV=C CINT=5 POWER=5 DASH=NEG**
4. For the following exercises, you can use RTGRIDS/GFS to create a McIDAS grid, but grid availability changes, so we've created a reliable McIDAS Grid file to download from a FTP site to use for the following exercises. For this example, the GRID4002 file must be placed in the directory ***<local-path>/mcidas/data*** directory. (This will be the same directory where GRID4000 and GRID4001 from previous exercises have be written.)
   Download the file from ftp.ssec.wisc.edu.
   ftp ftp.ssec.wisc.edu
   login anonymous
   prompt
   binary
   cd pub/mug/Data/Grid
   get GRID4002
5. Create a plot on frame 1 of the 24 hour 700 mb height forecast over the U.S. from the model run in the local TG.3 dataset. Use a label size of six pixels, a 60 m contour interval and label every other contour. Switch to the frame after plotting. Note: To create your own grid file from current data, run the command: GRDCOPY RTGRIDS/GFS TG.3 DAY=#Y TIME=0 FHOUR=0 6 12 24 GPRO=MERC NUM=ALL DEL=YES
   Type: **GRDDISP TG.3 FHOUR=24 LEV=700 PARAM=Z MAP=USA PRO=MERC GRA=1 LSIZE=6 CINT=60 LINT=2 SF=YES**
6. Overlay the 500 mb temperatures in degrees C in yellow. Dash the contours and use a label size of six pixels and a contour interval of 5.
   Type: **GRDDISP TG.3 FHOUR=24 LEV=500 PARAM=T UNIT=C NAV=C LSIZE=6 COLOR=3 CINT=5 DASH=ALL**
7. Create a plot of wind barbs over the US at the 250 mb level using the TG.3 dataset. Draw the barbs in yellow with a size of 12 on frame 2 and show the frame after the plotting is complete.
   Type: **GRDDISP TG.3 FHOUR=24 LEV=250 PARAM=WINDB MAP=USA LSIZE=12 COLOR=3 GRA=2 SF=YES**
   Notice that the density of the grid points and the size of the wind barbs makes them overlap.

8. Erase the frame and redo the previous plot, this time plotting only every other grid point.
   Type: **ERASE G;GRDDISP TG.3 FHOUR=24 LEV=250 PARAM=WINDB MAP=USA LSIZE=12 COLOR=3 PINT=2 2**
   The PINT keyword can be used to declutter a dense grid plot.

9. Erase frames 1-4. Create a sequence of the 0, 6, 12, and 24 hour 500 mb height field over the US from the 0Z GFS model run.
   Type: **ERASE F 1 4**
   Type: **GRDDISP TG.3 TIME=0 FHOUR=0 6 12 24 LEV=500 PARAM=Z CINT=60 MAP=USA GRA=1 4 LSIZE=6 LINT=2 SF=YES**
10. Set the loop bounds to frames 1-4. Change the dwell rate so it pauses longer on the last frame. Then, start the loop.
    Type: **LS 1-4;DR 3*3 10**
    Press: **Alt L**
11. Stop the loop.
    Press: **Alt L**

In this exercise, you will display the forecasted sea level pressures for the GFS 0Z model run in TG.3. Then, you will compute the 1000-500 mb thickness for that same time and display the thickness on top of the sea level pressure contours.

1. List the available sea level pressures from the 0Z model run.
   Type: **GRDLIST TG.3 NUM=ALL PAR=P LEV=MSL**
2. Erase the graphics in the first four frames.
   Type: **ERASE G 1 4**
3. Display the sea level pressure data from the 0Z model run. Plot the contours over a map of the United States.
   Type: **GRDDISP TG.3 FHOUR=24 PAR=P LEV=MSL MAP=USA**
4. Calculate and display the 1000-500 mb thickness for the same forecast time.
   Type: **GRDDISP TG.3 MAP=USA FHOUR=24 MATH='G2-G1' COLOR=3 NEWPAR=THCK GPM G1='LEV 1000;PAR Z' G2='LEV 500'**
5. Exit McIDAS.
   Type: **EXIT**

String Tables

Table of Contents:

String Tables

Strings are often used as shorthand for typing commonly used commands, parts of commands, or non-executable text, and for initializing your workstation during logon. For example, if you often load satellite images centered around latitude 43:35:15 and longitude 98:25:40, you could assign the string name LL to the string 43:35:15 98:25:40. Then whenever you would normally enter the latitude and longitude coordinates, you could now use the string LL. For example, instead of typing:

> **IMGDISP** *dataset* *frame* **LATLON=43:35:15 98:25:40**

you could type:

> **IMGDISP** *dataset* *frame* **LATLON=#LL**

This lesson describes how to:

- create and use strings
- edit strings
- save strings and string tables
- restore string tables
- delete strings and string tables
- increment values in a string

The following commands are used in this lesson.

Command	Function
ASK1	defines questions to prompt the user for string input
REPEAT	repeats a McIDAS command while incrementing command values
TD	deletes strings from the string table
TE	enters a string into the string table
TL	lists the strings in a string table
TU	saves, deletes, lists, and restores string tables

String Tables - Basic Concepts

Creating Strings

Strings are created by assigning a string name to a character string with the TE or ASK1 command. A character string consists of alphanumeric and printable characters. When creating strings, keep in mind the following guidelines:

- String names can be up to 12 alphabetic characters and must be uppercase.
- Strings H and Y contain the current time and date and cannot be redefined or deleted.
- X cannot be used as a string name because McIDAS interprets it as a placeholder for default values in a command.
- String names beginning with a question mark (?) are global strings and cannot be deleted the same way as other strings.
- An expanded string definition can't be more than 160 characters.
- Any executable string must be uppercase.

Using Strings

To run a string containing a command, you must type a pound key (#) followed by the string name. For example, to run the string DISPIMAGE, which contains the command IMGDISP G16FD 4 LATLON=35 98 BAND=2, you would type #DISPIMAGE on the command line. To use a string containing part of a command, type the command and substitute #*stringname* at the appropriate position. For example, to use the string LL which contains the text 43:35:15 98:25:40, you could run the command IMGDISP G16FD 1 BAND=2 LATLON=#LL.

Strings can be nested, allowing one string to use another string. For example, the string DISPIMAGE defined as IMGDISP G16FD 1 BAND=2 LATLON=#LL has a nested string (LL). In the string table, #LL is expanded and DISPIMAGE is listed as IMGDISP G16FD 1 BAND=2 LATLON=43:35:15 98:25:40. The nested string must be defined before it is used in another string.

If the string contains a value that will change, an additional pound sign allows the string to be expanded when it is run. For example, if LL will change, type a pound sign before LL so the string DISPIMAGE becomes IMGDISP G16FD 1 BAND=2 LATLON=##LL. When the string DISPIMAGE is entered in the string table, it will appear as IMGDISP G16FD 1 BAND=2 LATLON=#LL. When DISPIMAGE is run, the current value of LL is inserted.

You can also assign strings to a single letter. To run these strings, press the Ctrl key and the appropriate letter. Strings can also be assigned to the function keys (F keys) with an Alt, Shift, or Ctrl combination. This is useful if you enter certain commands often.

Using Global Strings

A string whose first character is a question mark (?) is called a global string. Global strings are useful for defining strings you don't want accidently deleted. They are not saved in user-defined string tables, but remain in the string table until you delete them.

Deleting Strings and Global Strings

The TD command deletes individual strings, strings with a common prefix, or all the strings in the string table. When you delete all the strings in the string table, two system strings (H and Y) and any global strings remain. Global strings are stored in the string table and will remain unless you delete them using TD ALL GLOB.

Some McIDAS commands write global strings to the string table. If you have problems using the string table, clear out your string table with the TD ALL GLOB command.

Saving and Restoring String Tables

Each McIDAS session has its own string table which stores strings until they are saved into a user-defined string table with the TU SAVE command. The project number and the initials of the user who created it are saved with each user-defined string table. A string table can contain up to 256 strings, a workstation can store up to 512 string tables. When a user-defined string table is restored with the TU REST command, a copy of the strings from the user-defined string table is written into the string table. When a string table is saved, any existing table with the same name and user initials is overwritten.

To restore a user-defined string table within a McIDAS session, specify its name and user initials with the TU REST command. You cannot delete another user's string tables. When a string table is restored, the default table is overwritten and unsaved strings are lost (except global strings).

Determining the Current String Table

The strings contained in the string table depend on how the McIDAS session was started. If a McIDAS session was started:

- without logging on to the workstation, the string table is the one used in the previous session.
- with the **LOGON** *initials project* **X** command, the string table is the one used in the previous session.
- with the **LOGON** *initials project* **I** command, the string table is cleared and any unsaved strings are removed from the string table except global strings.
- with **LOGON** *initials project string*, the string table that matches *string* is restored and all strings that start with the table name are automatically run.

- with **LOGON** *initials project* . , a string table matching the user initials is restored, and all strings that start with the table name are automatically run.

Incrementing String Values

The REPEAT command increments variables in a command by running it multiple times until a specific value is reached. For example, you can use the REPEAT command with a string containing

---

String Tables - Listing, Creating and Using Strings

In this exercise, you will create several strings and run them.

1. Start a McIDAS session.
   At the Unix prompt:
   Type: **`mcidas`**
   Your session should still be set for twenty frames from the last time you edited the Configuration GUI. If not, set it for six frames, as described in Getting Started.
2. List the strings in the string table.
   Type: **TL**
   TL without any parameters lists the first 20 strings that begin with a letter or number. The string table contains the H and Y system strings, which contain the current date and time, as shown in the example below.
   ```
   TL H := 19:09:02 Y := 2020231 --END OF LIST
   ```
3. Position the cursor in the center of the frame and run the Y string with the ZA command to print the date on frame 1.
   Type: **PC C;ZA "#Y**
   Notice that the string is expanded in the text window; the current Julian date (*ccyyddd*) replaces #Y.
4. Define a string called NOTE that stores a comment.
   Type: **TE NOTE "MAP OF THE WORLD**
5. Define a string called WORLDMAP that contains the command MAP WORL 3 BOX=NO.
   Type: **TE WORLDMAP "MAP WORL 3 BOX=NO**
6. Define a global string called CROSS to create a green cross-hair cursor.
   Type: **TE ?CROSS "CUR 51 51 XBOX GREEN**
   A string whose first character is a question mark (?) is defined as global. Global strings remain in the table unless deleted explicitly with TD ALL GLOB.
7. List the strings in the string table and erase frame 1.
   Type: **TL;ERASE G 1**
   The string table now contains four strings: H, NOTE, WORLDMAP, and Y, as shown below.
   ```
   TL H := 19:16:43 NOTE := MAP OF THE WORLD WORLDMAP := MAP WORL 3 BOX=NO Y := 1999362 --END OF LIST
   ```
   Note that the global string ?CROSS did not get listed.
8. Now list all of the strings in the current table, including the global strings.
   Type: **TL OUT**
   ?CROSS should be included in the listing.
9. Display a map by running the string called WORLDMAP.
   Type: **#WORLDMAP**
   A map of the world is drawn on the frame.
10. Change the cursor by running the CROSS string.
    Type: **#?CROSS**
11. Position the cursor at TV line 15 and element 260 and use the string NOTE with the ZA command to print text on the screen.
    Type: **PC T 15 260;ZA "#NOTE**
    The text MAP OF THE WORLD is printed on the screen.
12. Erase frame 1.
    Type: **ERASE G 1**
13. Define a string name T to print the current time. Remember that the ZA command requires a double quote before text to be printed.
    Type: **TE T "ZA "#H**
    Notice the expanded string in the text window. The system time when the string was created (HH:MM:SS) replaces the #H as shown below.
    ```
    T := ZA "20:12:07
    ```
14. Run the string called T. Move the cursor and run the command several times. Because the string name is a single letter, you can press Ctrl and the letter to run the string.
    Press: **Ctrl T**
    The time is drawn on the screen. Notice that it is the same each time the string is run.
15. Define the string CT so that the time updates to the current time each time the string is run. The double pound signs indicate that the value of string H should not be replaced until string CT is executed.
    Type: **TE CT "ZA "##H**
    Notice the expanded string in the text window as shown below.
    ```
    CT := ZA "#H
    ```
16. List the strings in the string table and notice the difference between T and CT.
    Type: **TL**

The #H string is expanded in the T string and not in the CT string.

```
TL CT := ZA "#H H := 20:42:30 NOTE := MAP OF THE WORLD T := ZA "20:12:07 WORLDMAP := MAP WORL 3 BOX=NO Y := 1999362
--END OF LIST
```

17. Move the cursor and run the string called CT. Move the cursor and run the command several times.

    Type: **#CT**

    The time is drawn on the screen; it is updated each time the string is run.

18. Define a string using the F4 function key.

    Type: **TE KEYF4 "ZA "USA**

19. Move the cursor and run the string called KEYF4. Move the cursor and run the command several times.

    Press: **F4**

String Tables - Editing Strings

In this exercise, you will use the ASK1 command to edit a string from the previous exercise and to prompt the user to create new strings.

1. List strings in the string table.
   Type: **TL**
   You should see five user-defined strings (CT,KEYF4,NOTE,T,WORLDMAP) and two system strings (H,Y) as shown below.
   ```
   TL CT := ZA "#H H := 20:42:30 KEYF4 := ZA "USA NOTE := MAP OF THE WORLD T := ZA "20:12:07 WORLDMAP := MAP WORL 3
   BOX=NO Y := 2020231 --END OF LIST
   ```
   TL without any parameters lists the first 20 strings in the table that start with a number or a letter.

2. Change the NOTE string to read "MAP OF THE ENTIRE WORLD!"
   Type: **ASK1 NOTE**
   In the pop-up window, edit the text to read "MAP OF THE ENTIRE WORLD!"

3. Check that the NOTE string was changed.
   Type: **TL NOTE**
4. Prompt the user to input a state and a four-digit METAR station.
   Type: **ASK1 STATE 'Input a 2-character state' STATION 'and a 4 digit METAR station'**
   In the pop-up window, enter a two-character state (e.g. TX) and a four-digit station (e.g. KLAR).

5. Check that the STATE and STATION strings were added to the string table.
   Type: **TL STA**
   You should see the STATE and STATION strings listed.
   ```
   STATE := TX STATION := KLAR
   ```

String Tables - Saving Strings and String Tables

In this exercise, you will save the strings from the previous exercises to the user-defined string table LEARN.

1. List strings in the string table.
   Type: **TL**
   You should see seven user-defined strings (CT,KEYF4,NOTE,STATE,STATION,T,WORLDMAP) and two system strings (H,Y) as shown below.
   ```
   TL CT := ZA "#H H := 20:42:30 KEYF4 := ZA "USA NOTE := MAP OF THE ENTIRE WORLD! STATE := TX STATION := KLAR T := ZA "20:12:07 WORLDMAP := MAP WORL 3 BOX=NO Y := 2020231 --END OF LIST
   ```
   TL without any parameters lists the first 20 strings in the table that start with a number or a letter.

2. List all the strings in the string table.
   Type: **TL OUT**
   Using the OUT parameter lists all strings in the string table, including the global string, ?CROSS.

3. Save the strings you created in a string table named LEARN.
   Type: **TU SAVE LEARN**
   A copy of each string is saved in the user-defined string table LEARN. The strings also remain in the current string table until you delete them or restore another user-defined string table.

4. List all user-defined string tables on your workstation.
   Type: **TU LIST**
   You should see a string table named LEARN with your initials (USER) associated with it.
   ```
   # NUM SAVE NAME INIT PROJ SAVED RESTORE USER --- --- ------------ ---- ---- ------- ------- ---- 1 1 LEARN USER 2020231
   ```

5. Delete all the strings in the string table.
   Type: **TD ALL**

   TD ALL removes the regular strings from the current string table, but does not remove the global or system strings.

   Also, when you saved the strings into the user-defined string table LEARN, a copy of each string was stored in LEARN and a copy remained in the string table. When you deleted the strings, they were removed from the current string table, but not from the user-defined string table LEARN.

String Tables - Restoring String Tables

1. In the last step of the previous lesson, you deleted all the strings in the string table. Now, list all the strings in the current string table,
   Type: **TL OUT**
   and verify that only the global strings and the H and Y system strings remain.

2. Restore the string table LEARN.
   Type: **TU REST LEARN**
   A copy of the strings in LEARN are placed in the string table.

3. List all the strings in the string table.
   Type: **TL OUT**
   The strings that you saved in the user-defined string table LEARN were restored to the string table.

String Tables - Deleting Strings and String Tables

In this exercise, you will list strings in the string table, delete them, then delete the entire string table.

1. List the strings in the string table that begin with the letter N.
   Type: **TL N**
   The string NOTE is listed.

2. Delete the string called NOTE.
   Type: **TD NOTE**
3. List the strings in the string table to see if NOTE was deleted.
   Type: **TL OUT**
   Also note that the global string ?CROSS is listed. You will delete ?CROSS in the next step.

4. Delete all the strings that begin with a question mark (?). This will delete all of your global strings.
   Type: **TD PREFIX ?**
5. List the strings in the string table again to verify that the global strings were deleted.
   Type: **TL OUT**
6. Delete all the strings that begin with the letter T.
   Type: **TD PREFIX T**
7. List the strings in the string table again.
   Type: **TL OUT**
   Your string table should contain seven strings: two system strings, H and Y, and the user-defined strings CT, KEYF4, STATE, STATION and WORLDMAP.

8. Delete all the strings in the string table, then list the strings to verify they were deleted.
   Type: **TD ALL;TL OUT**
9. List the string tables on your workstation.
   Type: **TU LIST**
10. Restore the user-defined string table LEARN.
    Type: **TU REST LEARN**
11. List the strings. You will see the strings from the user-defined string table LEARN have been restored.
    Type: **TL OUT**
12. Delete the string table called LEARN.
    Type: **TU DEL LEARN**
13. Verify that the user-defined string table LEARN was deleted, and see that the strings in the string table remain.
    Type: **TU LIST;TL OUT**

String Tables - Incrementing Values in a String

In this exercise, you will define a simple string that displays three images in three consecutive frames, use the REPEAT command to define values for the string, and run the string.

1. Create a string called DISPLAY that displays images on frames.
   Type: **TE DISPLAY "IMGDISP G16C.!1 !2 STA=DCA BAND=13**
   The exclamation point (!) indicates the value will be defined and incremented when you use the REPEAT command. In this example, the dataset position number (!1) and the frame number (!2) will increment.

2. Repeat the DISPLAY string three times, starting at G16C dataset position 1 and frame 4, and increase each value by one until dataset position 3 is displayed in frame 6.
   Type: **REPEAT DISPLAY 1 TO 3 BY 1 4 BY 1**
3. Set the loop bounds to frames 4 through 6 and loop the frames. Stop looping.
   Type: **LS 4-6**
   Press: **Alt L**
4. Exit McIDAS
   Type: **EXIT**

Real-time Data Access

Table of Contents:

Real-time Data Access

So far, this Learning Guide has introduced you to McIDAS and given you the basic tools and concepts needed to use the McIDAS system. The lessons have used data from local data files and real-time sources. This lesson will focus on all types of real-time meteorological data.

Through the real-time servers at SSEC, weather text products from the National Weather Service are available, as well as conventional grid data, and surface and upper-air point data. SSEC also receives satellite imagery from both polar orbiting and geostationary satellites. Throughout the Learning Guide you have used data servers that contain freely available real-time data via Unidata and collaborative efforts between SSEC and Unidata.

The examples listed in this chapter assume that you are using the same dataset names as those used at SSEC. If your dataset names are different, you may have to change the dataset names in some of the commands.

This lesson describes how to:

- set up your routing table to access real-time data
- utilize McIDAS applications that use raw text data
- plot and list weather watches and warnings
- display current and forcasted fronts over maps and satellite data
- easily list and display real-time surface observations and upper-air data
- create Skew-T thermodynamic diagrams, vertical cross sections, and meteorograms
- display and manipulate real-time satellite images
- display real-time NEXRAD images
- access real-time gridded model data

The following commands are used in this lesson:

Command	Function
DATALOC	manages and lists the client routing table
DSINFO	lists ADDE datasets on local and remote servers
FRMLIST	lists frame directory information
FRNTDISP	displays fronts from NCEP frontal positions products
GEO	finds and displays a geostationary satellite image over a point or region
GRDDISP	generates contours, streamlines or plots from grid data
GRDINFO	lists, modifies, or statistically analyzes grid point values
GRDLIST	lists grids in a dataset
HODO	plots a hodograph of upper-air data
IMGDISP	displays image data
IMGLIST	lists images in a dataset

IMGPROBE	lists or plots image data for a region of the displayed image
IMGREMAP	remaps images into different projections
LEO	finds and displays a polar orbiting satellite image over a point or region
MAP	displays a map and latitude/longitude lines on a graphics frame
MOSRPT	lists MOS forecasts for the ETA, GFS and NGM models
PTLIST	lists point data in a dataset
RAOBCON	contours upper-air data from IRAB-schema point files
RAOBPLOT	plots upper-air data from IRAB-schema point files
RAOBRPT	lists upper air (RAOB) observations
READ	lists the contents of an ADDE text dataset
SEE	lists the text in an LW file
SFCCON	contours surface hourly data from ISFC-schema point files
SFCLIST	lists surface hourly data from point files
SFCMG	displays a surface meteorogram
SFCPLOT	plots surface hourly data from ISFC-schema point files
SFCRPT	lists surface hourly observations
STNLIST	lists weather stations
UACROSS	plots a vertical cross section of upper-air data
UALIST	lists upper-air data
UAPLOT	plots a sounding on a thermodynamic diagram
WWDISP	displays active weather warnings, watches and advisories
WWLIST	lists active weather warnings, watches and advisories
WXTLIST	lists weather text products

Real-time Data Access - Getting Started

The exercise below will take you through the steps necessary for setting up your client routing table to access real-time data. Once you have real-time groups included in the client routing table, ADDE commands can access data as soon it becomes available on the server machine. These exerises will use free data servers available from Unidata and the Unidata/SSEC collaboration.

Users who have LOGON credentials (initials and project number) that allow access to the SSEC real-time servers can modify the commands to use those datasets. Remember that once you have your client routing table set up to access real-time data from SSEC, any requests you make for real-time data will be charged to the project number you use when you log on. Run the following commands to setup access to the SSEC servers that use LOGON credentials: **LOGON** *initials project#* **I; DATALOC ADD INFO INFO.SSEC.WISC.EDU; READ INFO/RTSERVER DEV=T RTSERVER.BAT R; BATCH RTSERVER.BAT**

Users who **do not** have access to the SSEC real-time servers and would prefer to use their own site's data sources should contact their Site Coordinator (System Administrator) for instructions on how to set up the client routing table. Once you have your table set up, you may start your McIDAS session and go to the next section. Remember, this chapter assumes that you are using the same dataset names as those used at SSEC. If your dataset names are different, you may have to change the dataset names in some of the commands.

1. Start a McIDAS session.
   At the Unix prompt:
   Type: **`mcidas`**
   Your session should still be set for twenty frames from the last time you changed the Configuration GUI. If not, set it for twenty frames, as described in Getting Started.
2. Modify your client routing table so that commands are requesting data from freely accessed image, grid, point and text datasets.
   Type: **DATALOC ADD RTGOESR ADDE.UCAR.EDU**
   Type: **DATALOC ADD RTGOESS ADDE.UCAR.EDU**
   Type: **DATALOC ADD RTGRIDS ADDE.SSEC.WISC.EDU**
   Type: **DATALOC ADD RTPTSRC ADDE.SSEC.WISC.EDU**
   Type: **DATALOC ADD RTWXTEXT ADDE.SSEC.WISC.EDU**
3. List your client routing table's groups and associated server IP addresses.
   Type: **DATALOC**
   All of the real-time servers should be displayed, in addition to any group names that were previously set up on your workstation.

4. Run the DSINFO command to list all datasets of type TEXT in the group RTWXTEXT.
   Type: **DSINFO TEXT RTWXTEXT**
   A listing of all of the descriptor names in the group RTWXTEXT is displayed. The contents of these text datasets will be used by commands that use these datasets for data access.

```
Dataset Names of Type: TEXT in Group: RTWXTEXT Name NumPos Content ------------ ------
-------------------------------------- FOUS14 1 Real-Time FOUS14 Text GFSMOS 1 Real-Time GFS Model Output Statistics
NAMMOS 1 Real-Time NAM Model Output Statistics SFCHOURLY 1 Real-Time SFC Hourly Text SYNOPTIC 1 Real-Time SYNOPTIC
Text TERMFCST 1 Real-Time Terminal Forecast UPPERAIR 1 Real-Time Upper Air Text
```

5. Run the DSINFO command to list all the datasets available in the RTGRIDS group.
   Type: **DSINFO X RTGRIDS**
   The X is a placeholder that will list all of the data types in RTGRIDS. The DSINFO command can be used with any of the groups listed in your client routing table.

Real-time Data Access - Weather Text Products

Weather text data is unique, because it must reside on a McIDAS-X remote server; it cannot reside on the local server.

The first line of text products received by McIDAS contains WMO information about the data being received. An example of this WMO information line is shown below:

Example: FPUS5 KMKE 171245

Format: *product origin date*

product the product header; includes a two-character WMO product code (FP), a two-character geographic identification code (US), and an optional product ID number between 0 and 99 (5)

origin the station initiating the report (KMKE)

date day of the month (17), hour UTC (12), and minute (45)

Many text products sent by the National Weather Service contain an additional line of information. This AFOS/AWIPS information also indicates product and origin. Examples of the AFOS/AWIPS information lines are shown here:

Examples: ZFPWI or LFPMKE

Format: *pppsss*

ppp the product header; three-character AFOS/AWIPS product code (ZFP or LFP)

sss the station initiating the report; two or three characters, depending on the AFOS product (WI or MKE)

Using WXTLIST to read weather text products

In this lesson, you will be using the WXTLIST command to read various weather text products.

1. List the available predefined text products.
   Type: **WXTLIST DIR**
   This will produce a listing of all the predefined text products available in the group RTWXTEXT.

2. List the five most recent Severe Weather Statements, using one of the predefined text products.
   Type: **WXTLIST SEVERE_WX_STATEMENT NUM=5**
   WXTLIST relies on keywords to refine the search for text products. The NUM keyword indicates how many reports you want listed. The WSTN keyword can be used to match the station initiating the report.

3. List the most recent zone forecast for Madison, Wisconsin.
   Type: **WXTLIST APRO=ZFP ASTN=MKX MATCH=MADISON**
   The APRO keyword matches the product header. The ASTN keyword matches the station initiating the report. The MATCH keyword looks for any occurrence of the word MADISON in the reports.

Listing raw observational text reports

There are five macro commands that list observations and forecasts:

- MOSRPT (to list MOS forecasts from the GFS, NAM and NGM models)
- RAOBRPT (to list upper air observations)
- SFCRPT (to list surface hourly observations)
- SYNRPT (to list synoptic observations)
- TAFRPT (to list terminal aerodrome forecasts)

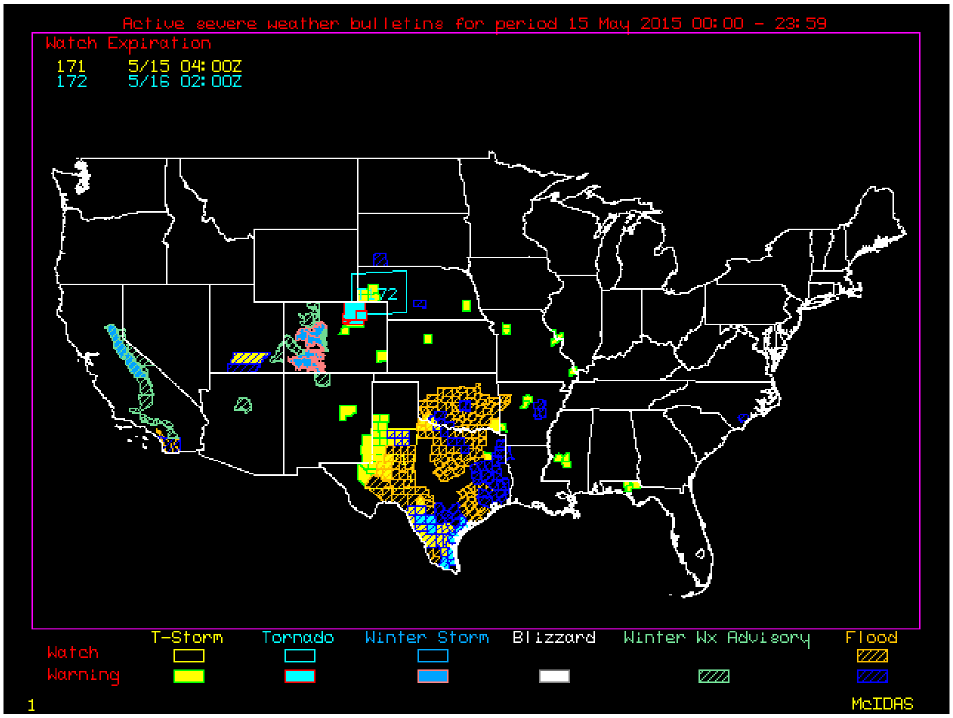
These commands all utilize the functionality of the OBSRPT command.

1. List the upper air observations for Arizona and New Mexico for the past 24 hours.
   Type: **RAOBRPT AZ NM 24**
2. List the surface hourly observations for the past 12 hours in Phoenix, Arizona.
   Type: **SFCRPT KPHX 12**
   This data is all listed in the raw report format. In a later section, you will use the UALIST and SFCLIST commands to quickly list the observed data in a format that is easier to read.

Real-time Data Access - Watches and Warnings

In this exercise, you will list and display severe thunderstorm, tornado and winter storm watches and warnings, as well as blizzard warnings, snow advisories, and flood watches and warnings.

1. Display a map of the United States with all of todays watches, warnings and advisories plotted.
   Type: **WWDISP USA TIME=0 23:59**
   Sample output from WWDISP



2. Find a state that has a box plotted and list the reports for that state (in this case, Mississippi).
   Type: **WWLIST MS TIME=0 23:59**
   Replace MS with whatever state you would like listed. The list should look something like the following:

```
WWLIST MS TIME=0 23:59 Severe Thunderstorm Warning valid until 15 May 2015 1915UTC for the following counties
MS:SIMPSON Severe Thunderstorm Warning valid until 15 May 2015 2015UTC for the following counties MS:HINDS MS:RANKIN
WWLIST: done
```

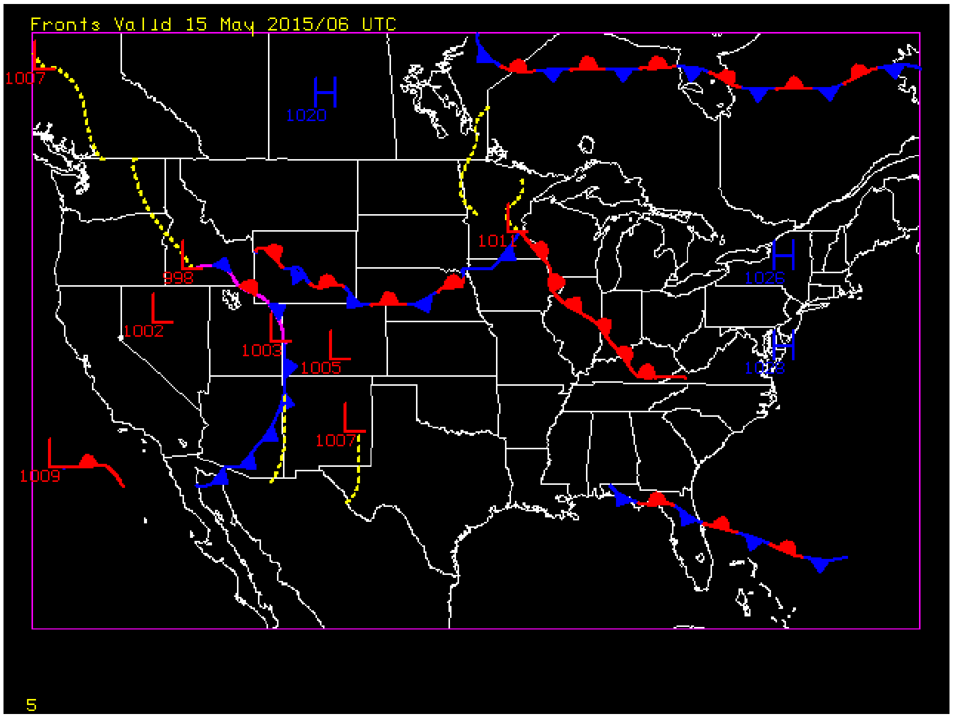
3. Output the actual reports for the state.
   Type: **WWLIST MS TIME=0 23:59 OUT=RAW**
   This will output the actual textual report that was issued by the National Weather Service. The output may look like the following:

```
WWLIST MS TIME=0 23:59 OUT=RAW WUUS54 KJAN 151828 2015135 1829 SVRJAN MSC127-151915-
/O.NEW.KJAN.SV.W.0106.150515T1828Z-150515T1915Z/ BULLETIN - IMMEDIATE BROADCAST REQUESTED SEVERE THUNDERSTORM
WARNING NATIONAL WEATHER SERVICE JACKSON MS 128 PM CDT FRI MAY 15 2015 THE NATIONAL WEATHER SERVICE IN JACKSON HAS
ISSUED A * SEVERE THUNDERSTORM WARNING FOR... EASTERN SIMPSON COUNTY IN CENTRAL MISSISSIPPI... * UNTIL 215 PM CDT *
AT 128 PM CDT...A SEVERE THUNDERSTORM WAS LOCATED OVER SANATORIUM...OR NEAR MAGEE...MOVING NORTH AT 30 MPH.
HAZARD...60 MPH WIND GUSTS AND QUARTER SIZE HAIL. SOURCE...RADAR INDICATED. IMPACT...HAIL DAMAGE TO VEHICLES IS
EXPECTED. EXPECT WIND DAMAGE TO ROOFS...SIDING AND TREES. * THIS SEVERE THUNDERSTORM WILL BE NEAR... MENDENHALL AND
MARTINVILLE AROUND 145 PM CDT. OTHER LOCATIONS IMPACTED BY THIS SEVERE THUNDERSTORM INCLUDE BRAXTON AND D'LO.
PRECAUTIONARY/PREPAREDNESS ACTIONS... FOR YOUR PROTECTION MOVE TO AN INTERIOR ROOM ON THE LOWEST FLOOR OF A
BUILDING. && LAT...LON 3205 8998 3205 8966 3178 8967 3176 8994 TIME...MOT...LOC 1828Z 174DEG 25KT 3186 8979
HAIL...1.00IN WIND...60MPH $$ WUUS54 KJAN 151928 2015135 1928 SVRJAN MSC049-121-152015-
/O.NEW.KJAN.SV.W.0107.150515T1928Z-150515T2015Z/ BULLETIN - IMMEDIATE BROADCAST REQUESTED SEVERE THUNDERSTORM
WARNING NATIONAL WEATHER SERVICE JACKSON MS 228 PM CDT FRI MAY 15 2015 THE NATIONAL WEATHER SERVICE IN JACKSON HAS
ISSUED A * SEVERE THUNDERSTORM WARNING FOR... SOUTHWESTERN RANKIN COUNTY IN CENTRAL MISSISSIPPI... SOUTHEASTERN
HINDS COUNTY IN CENTRAL MISSISSIPPI... * UNTIL 315 PM CDT * AT 227 PM CDT...A SEVERE THUNDERSTORM WAS LOCATED NEAR
WHITES...OR NEAR BYRAM...MOVING NORTH AT 10 MPH. HAZARD...60 MPH WIND GUSTS AND QUARTER SIZE HAIL. SOURCE...RADAR
INDICATED. IMPACT...HAIL DAMAGE TO VEHICLES IS EXPECTED. EXPECT WIND DAMAGE TO ROOFS...SIDING AND TREES. * THIS
SEVERE THUNDERSTORM WILL BE NEAR... BYRAM AROUND 250 PM CDT. RICHLAND AROUND 300 PM CDT. JACKSON AROUND 315 PM CDT.
PRECAUTIONARY/PREPAREDNESS ACTIONS... FOR YOUR PROTECTION MOVE TO AN INTERIOR ROOM ON THE LOWEST FLOOR OF A
BUILDING. && LAT...LON 3206 9003 3205 9029 3226 9039 3232 9009 TIME...MOT...LOC 1927Z 163DEG 10KT 3213 9021
HAIL...1.00IN WIND...60MPH WWLIST: done
```

Real-time Data Access - Fronts

In this exercise you will display current and forecasted NCEP fronts on maps and over satellite images.

1. Erase the images and graphics on frames 1 through 4.
   Type: **ERASE F 1 4**
2. Display the most recent NCEP frontal positions on a map of the United States.
   Type: **FRNTDISP USA LATEST**
   Entering 'LATEST' in the time parameter will use the latest analysis within the previous 6 hours. You should get a map that looks similar to this:



3. Next, list the decoded front point and High/Low positions in a tabular format.
   Type: **FRNTDISP USA LATEST OUT=TAB**
   You will see that the data is listed in a format that labels the latitude, longitude and pressure of each High and Low, and the latitude and longitude for each point of the troughs and fronts.

4. Display the most recent GOES-East IR Full Disk image centered over Topeka with a magnitude of -2 on frame 2. Overlay the latest NCEP fronts on the image.
   Type: **SF 2;IMGDISP RTGOESR/FD BAND=13 MAG=-2 STA=TOP;MAP**
   Type: **FRNTDISP**
   The default of FRNTDISP is to use the current frame navigation to display the latest NCEP frontal positions.

5. Display the 24 hour forcast over the United States at 0Z on frame 3.
   Type: **SF 3;FRNTDISP USA 0 FHOUR=24**
   This commands draws the 24-hour forcast fronts in today's 00 UTC NCEP frontal position analysis product on a map of the United States.

Real-time Data Access - Upper-air and Surface Observations

In a previous lesson, you used PTLIST, PTDISP, and PTCON to list, plot and contour data from point source files. These commands are very flexible and can access any type of data stored in MD files. However, SFCCON, SFCPLOT, SFCLIST, RAOBCON, RAOBPLOT, and UALIST are "macro-like" commands that build and run a PTLIST, PTDISP or PTCON command to access surface or upper-air point data.

These short commands are easier to use with real-time data, because the keyword defaults are set up to easily access the most recent data available. So, when you are using these commands, you will not have to specify the DATASET keyword, and you will only have to specify TIME or DAY if you want to look at something other than the most recent real-time data.

In this example, you will list real-time point data using SFCLIST and UALIST, and then you will use RAOBCON, RAOBPLOT, SFCCON, and SFCPLOT to contour and plot real-time surface and upper-air point data.

Listing real-time point data

1. List the current real-time surface data for the state of Hawaii.
   Type: **SFCLIST HI**
   This command defaults to listing data from the current day and time using the RTPTSRC/SFCHOURLY dataset.

2. Run the same command, but this time sort the stations in chronological order, instead of in alphabetical order.
   Type: **SFCLIST HI SORT=OBTIME**
3. List the current real-time upper-air data for all of the RAOB stations in the state of Texas.
   Type: **UALIST ST=TX**

Displaying and contouring real-time point data

1. Erase both the images and graphics in frames 1 through 4.
   Type: **ERASE F 1 4**
2. Display the two latest GOES-East IR images on frames 1 and 2 centered on Charlotte, North Carolina. Show each frame and add a map in graphics color level 2.
   Type: **IMGDISP RTGOESR/CONUS BAND=13 MAG=-2 ALL=1 2 STA=CLT SF=YES REFRESH='MAP H 2'**
3. Show frame 1 and plot today's 00 UTC 850 mb heights on the satellite image using data from the real-time mandatory upper air point data set. Plot the height values in incrementing color values every 20m between 1450m and 1550m. Start with color level 2.
   Type: **SF 1;RAOBPLOT Z 850 SAT 00 COLOR=2[1450-1550B20]**
   You do not need to specify the DATASET keyword or the DAY positional parameter. Since you are interested in today's data from the RTPTSRC/UPPERMAND dataset, you can just use the defaults for DAY and DATASET. The COL= keyword will plot values 1450-1469 in level 2 (cyan), values 1470-1489 in level 3 (yellow)... values 1530-1549 in level 6 (blue), and value 1550 in level 7 (white). Values less than 1450 are in level 1 (magenta), and values greater than 1550 are in level 8 (gray).

4. Draw contours over the plot in frame 1 using the same data. Draw the contours in graphics color level 5, with a contour interval of 30.
   Type: **RAOBCON Z 850 SAT 00 COLOR=5 CINT=30**
5. Show frame 2 and plot the current weather symbols on the satellite image using data from the RTPTSRC/SFCHOURLY dataset (the default dataset). Draw the symbols in graphics color level 4.
   Type: **SF 2;SFCPLOT WXS COLOR=4**
6. Contour the current temperature over the plot using the same data and conditions as in the previous step. Define a contour interval of 5 degrees and draw the contours in color 3.
   Type: **SFCCON T COLOR=3 CINT=5**
7. Plot the winds over South Carolina at 00 UTC today in frame 3. Use data from the surface (the default) and ship point data files and draw the barbs in green (color 4).
   Type: **SF 3;SFCPLOT WINDB SC 00 COLOR=4 DATA=RTPTSRC/SFCHOURLY RTPTSRC/SHIPBUOY**
8. Draw the 00 UTC 200 mb streamlines over the United States in graphics color level 5. Show the frame when the plot is finished.
   Type: **RAOBCON STREAML 200 USA 00 COLOR=5 GRA=4;SF 4**

Real-time Data Access - Meteorological Diagrams

In this exercise, you will create four types of meteorological diagrams: a Skew-T diagram, a hodograph, a vertical cross section and a meteorogram. You will also list surface data in a tabular format.

1. Erase frames 1 through 6.
   Type: **ERASE F 1 6**
2. List all of the upper-air stations reporting a 00 UTC observation today. The output is sorted alphabetically by state and then by ID.
   Type: **STNLIST CO=US DAT=RTPTSRC/UPPERMAND TIME=00**
3. Choose one of the stations listed. Show frame 1 and plot a Skew-T diagram for your chosen station (in this case, 72645, which is Green Bay, Wisconsin) at 00 UTC.
   Type: **SF 1;UAPLOT 72645 00**
4. Run the same command, but this time activate the interactive Skew-T.
   Type: **UAPLOT 72645 00 EDIT=YES**
   Follow the instructions in the command output in the text window to change the temperature or dewpoint sounding. Notice the sounding analysis parameters change when you make the atmosphere more or less stable.

5. Show frame 2 and plot a hodograph of the same station.
   Type: **SF 2;HODO 72645 00**
6. Show frame 3 and display a map of the upper-air stations reporting a 00 UTC observation.
   Type: **SF 3;RAOBPLOT IDN X MID 00**
7. Choose the two stations that are the farthest apart on the map. Show frame 4 and display a vertical cross section of theta, wind, and mixing ratio values from these two stations (in this case, International Falls, Minnesota (72747) to Nashville, Tennessee (72327). The MAP=YES keyword displays a map in the upper-right corner showing the cross section location and stations selected.
   Type: **SF 4;UACROSS 72747 72327 MAP=YES TIME=00 PAR=THA WINDB MIX**
8. Show frame 5 and plot a 24-hour surface meteorogram for Macon, Georgia.
   Type: **SF 5;SFCMG KMCN**
   The default dataset in the SFCMG command is RTPTSRC/SFCHOURLY, so the command automatically accesses the real-time data on the remote server. You do not need to have the data on your local machine.

9. List the surface hourly data for Macon, Georgia for the past 24 hours.
   Type: **SFCLIST KMCN 24**
   You should notice that the data listed in the Text Window is the same as that displayed in the Image Window.

Real-time Data Access - Satellite Imagery

In the first part of this exercise, you will access real-time satellite imagery and review many of the associated McIDAS commands. You will use these commands to display real-time satellite imagery, as well as list information about the data. A good review of satellite data and McIDAS AREA files can be found in the Basic Concepts section of the Satellite Imagery chapter.

Displaying Satellite Data

The DSINFO and IMGLIST commands list information about the images you have access to. Once located, you can display the images using the IMGDISP command. You can display images one at a time or in a sequence. An entire set of images can be loaded chronologically using one IMGDISP command.

1. List all datasets of type IMAGE in your routing table.
   Type: **DSINFO IMAGE**
   This will list all of the IMAGE datasets that you have access to in your routing table.

2. Erase graphics frames 1 through 4.
   Type: **ERASE G 1 4**
3. List the area directories for the four most recent GOES-East images.
   Type: **IMGLIST RTGOESR/FD.-3**
   ```
   Image file directory listing for:RTGOESR/FD Pos Satellite/ Date Time Center Band(s) sensor Lat Lon --- ------------- ------------ -------- ---- ---- ------------ 192 GOES-16 4 MAR 21063 18:40:16 0 75 1-16 191 GOES-16 4 MAR 21063 18:30:16 0 75 1-16 190 GOES-16 4 MAR 21063 18:20:16 0 75 1-16 189 GOES-16 4 MAR 21063 18:10:16 0 75 1-16 IMGLIST: done
   ```
4. Display the four most recent GOES-East IR images chronologically by using the ALL keyword. Place the images in frames 1 through 4 and center them on Miami.
   Type: **IMGDISP RTGOESR/FD BAND=13 ALL=1 4 STA=MIA**
   Because real-time satellite data is updated often, it is possible to view an image that has not yet been fully ingested by the server. If you attempt to access the most recent image in a dataset, you may get unexpected results. For GOES images, the image may be all black if the band requested isn't available yet on the server. This is normal behavior with GOES ABI imagery. When viewing older GOES GVAR images, you could see images filling the screen line-by-line as data was being ingested.

5. Set the loop bounds to loop through frames 1 through 4 and loop the frames. Stop the looping when you are finished looking at the images.
   Type: **LB 1 4**
   Press: **Alt L**
   Press: **Alt L**

Listing Image Information

When you display an image on a frame, McIDAS creates a frame directory for the image which lists the frame number, sensor source, date, image time, band number, image coordinates, magnification factors, and ADDE dataset name and file position number. The FRMLIST command lists the frame directory.

The IMGPROBE command lists data from the area represented on the image at the cursor center. It lists the area number, area coordinates, image coordinates, raw, brightness, and when present, temperature and radiance values. IMGPROBE can be invoked with its default parameters by pressing Alt-D.

Even though the data is not located on your local machine, but rather on some remote server, you can still access information about the data.

1. List the frame directory for all of the images in frames 1 through 4.
   Type: **FRMLIST 1 4**
   Information about the frame number, sensor source, band, date, time, and ADDE dataset/position number is displayed for each image.
   ```
   Frame Satellite [panel] Sensor Band Date Time Data Source --------- ------------ ---- ------------ -------- ----------------------------- 1[1] GOES-16 13 04 Mar 21063 18:40:16 RTGOESR/FD.189 2[1] GOES-16 13 04 Mar 21063 18:50:16 RTGOESR/FD.190 3[1] GOES-16 13 04 Mar 21063 19:00:16 RTGOESR/FD.191 4[1] GOES-16 13 04 Mar 21063 19:10:16 RTGOESR/FD.192
   ```
2. List the area values for the element at the center of frame one .
   Type: **SF 1;PC C;IMGPROBE MODE=N**
   Alternate method:
   Type: **SF 1;PC C**
   Press: **Alt D**
   The area number, area coordinates, image coordinates, raw, and brightness values are listed. Because this is an infrared image, the temperature and radiance values are also listed. (Note: It is possible a Frame/Image mismatch error will occur with the IMGPROBE or D command. This occurs when a new image becomes available on the server, and the absolute position number changes of the displayed image. You can redo the IMGDISP command in step 4 above and rerun the IMGPROBE/D commands to get the expected output.)
   ```
   Image Name Day Nominal Time Scan Time Band ---------------- ------- ------------ --------- ---- RTGOESR/FD.189 4 Mar 21063 18:40:16 18:42:19 13 File Nominal Image RAW RAD TEMP BRIT Lat/Lon Line/Element Line/Element * K 25:45:53/ 80:29:21 1364/ 2443 5457/ 9773 2258 101.61 297.86 64 * milliwatts/meter**2/steradian/(cm-1) IMGPROBE: Done
   ```

Real-time Data Access - Satellite Imagery (Continued)

In the next part of this exercise, you will be introduced to two new commands, which find and display geostationary and polar orbiting satellite images over a geographical area. You will also use IMGREMAP to manipulate real-time satellite images.

Displaying Satellite Images Using GEO and LEO

GEO and LEO are "macro-like" commands which build and run multiple McIDAS commands to access satellite data.

GEO finds and displays GEOstationary satellite images over a specified point or region, while LEO finds and displays polar orbiting (or Low Earth Orbit) satellite images. GEO uses the **GEO.***, **MAP.***, and **GEODATA.*** context files, where **.*** refers to **.CORE**, **.SITE**, and **.USER**. Default parameter and keyword values are found in the **GEO.*** context files, regional map definitions are found in the **MAP.*** context files, and satellite coverage, instrument channels, resolution information, and boarding image datasets are defined in **GEODATA.*** context files. LEO uses the same context files, except for a change in name to **LEO.*** and **LEODATA.***.

If you do not access your real-time data at SSEC, GEO and LEO may not work without changing the dataset names in the **GEODATA.*** and **LEODATA.*** files.

GEO and LEO have been designed so that the user does not need to know which satellite covers which earth location. The only thing the user needs to input is the geographical location that they are interested in. The commands find the applicable dataset(s) and display the images automatically.

This exercise will cover the GEO command using a **GEODATA.USER** file that contains the free datasets being used in the Learning Guide. (If you have access to SSEC's polar orbiting data servers, or have your own data servers for polar orbiting data, you can try using LEO with similar examples.)

1. List and read through the GEO help.
   Type: **HELP GEO**
2. Erase frames 1 through 4.
   Type: **ERASE F 1 4**
3. Download a **GEODATA.USER** that will allow the GEO command to access free GOES geostationary datasets. The GEODATA.USER file should be placed in the ***<local-path>/mcidas/data*** directory.
   Download the file from ftp.ssec.wisc.edu.
   ftp ftp.ssec.wisc.edu
   login anonymous
   prompt
   binary
   cd pub/mug/Data/Satellite
   get GEODATA.USER
4. Find and display a geostationary visible image centered over Torrance, California (KTOA) on frame 1.
   Type: **SF 1;GEO KTOA VIS**
   First, the command searches for the latitude and longitude of the location defined in the first positional parameter. Then, it uses **GEODATA.USER** to find the closest geostationary satellite to use. Next, if the user has not specified which channel to use, GEO will determine the channel, either visible or infrared, depending on the time of the day. Also, if the center point is at the edge of the satellite coverage, GEO will take the adjoining satellite and merge enough data to cover the rest of the image to create the composite. Finally, GEO will display the image with a map on the current frame.

5. Find and display a geostationary water vapor image centered over Vancouver, Canada using a filter. Display the image on frame 2.

   First, find the station ID for Vancouver, Canada.
   Type: **STNLIST CO=CA MATCH=VANCOUVER**
   ```
   IDN ID Station Name Data Types ST CO LAT LON ELE ----- ----- -------------------- ----------------- -- -- ---------- ---------- ---- 71042 ---- VANCOUVER HILLCREST 36 BC CA 49:13:48 123:06:00 84 71201 CWHC Vancouver Automatic M 36 BC CA 49:18:00 123:07:00 2 71784 CWWA W Vancouver Automati M 36 BC CA 49:21:00 123:11:00 178 71892 CYVR Vancouver Intl Air-P M 36 T BC CA 49:11:00 123:10:00 3 --- WHC VANCOUVER (AUTOB V BC CA 49:17:00 123:07:00 2 --- WVR VANCOUVER V BC CA 49:11:00 123:11:00 0 --- WWA W VANCOUVER AUTO V BC CA 49:21:00 123:10:00 178 --- YVR VANCOUVER/B.C. V BC CA 49:01:00 123:09:00 2 Number of stations listed: 8 STNLIST: Done
   ```
   After locating the station ID, use the GEO command to display the image.
   Type: **SF 2;GEO CYVR WV ON=FILTER TIME=0**
   Use the ON= keyword to set any of the four GEO options, such as FILTER, to on or off. The parameter WV indicates to display a water vapor image. TIME=0 is specified to select a desired time to display for the current day.

6. On frame 3, display the previous day's geostationary image centered over your home town from the current time.
   Type: **SF 3;GEO 'FORT ATKINSON[JEFFERSON],WI,US' DAY={enter previous day's date}**
   In the example above, the brackets **[ ]** indicate the county location. The DAY= keyword allows you to specify a day other than the current one. Valid entries for this keyword will depend on how many days of data are kept online in your servers.

7. Display the 12 UTC geostationary infrared image centered over Tahiti on frame 4.
   Type: **SF 4;GEO 'TAHITI,,PF' IR TIME=12 RES=5**
   This example displays the closest 12 UTC infrared image centered over Tahiti and displays it with a 5 km resolution. Note that two commas

appear between the city (Tahiti) and country code (PF) because if only one comma is specified, it's assumed to be a state or province code.

8. Display the most recent shortwave infrared image over the United States with no command output, a enhanced stretch to the image, and a yellow map.
   Type: **GEO USA SW OFF=ECHO ON=FILTER MCOLOR=3**
   This entry finds the satellites whose coverage includes the region defined by the map named USA in the **MAP.USER /.SITE /.CORE** file. It then displays the satellite's most recent image at the resolution needed to cover the region. For most users, the region will be the country of the United States because that's how USA is defined in the **MAP.CORE** file.

Manipulating Satellite Images

A major strength of McIDAS as a meteorological data analysis/display package is its ability to display and manipulate satellite images, even if the data is located on a remote server. With the IMGDISP command, you can change the image resolution displayed with the MAG= keyword. The IMGREMAP command allows you to remap images into new projections and create new image products.

1. Display the GOES-West Full Disk IR image in its original resolution (2 km) on frame 1. Center the image on earth coordinates 0° and 137°, and show frame 1 once the image is loaded.
   Type: **ERASE F 1 4**
   Type: **IMGDISP RTGOESS/FD 1 BAND=13 LAT=0 137 SF=YES;MAP X X LALO**
2. Decrease (blow down) the image resolution of the GOES-West Full Disk image by a factor of 5 and display it on frame 3.
   Type: **IMGDISP RTGOESS/FD 3 BAND=13 LAT=0 137 SF=YES MAG=-5;MAP X X LALO**
3. Magnify (blow up) the image resolution of the GOES-West Full Disk image by factor of 5 and display it on frame 2.
   Type: **IMGDISP RTGOESS/FD 2 BAND=13 LAT=0 137 SF=YES MAG=5;MAP X X LALO**
4. Change the loop sequence to view the images in the order of increasing resolution.
   Type: **LS 3 1 2**
5. Loop the frames.
   Press: **Alt L**
6. When finished viewing the loop, turn the looping off.
   Press: **Alt L**
7. Change the loop sequence back to loop through the first four frames.
   Type: **LB 1 4**
8. Remap the latest GOES-West IR image to a mercator projection centered on the Seattle-Tacoma International Airport, and place it in the fourth position in the MYDATA/IMAGES dataset. Change the output resolution, but only create a 480x640 sized image. Note: the MYDATA/IMAGES dataset was initially setup in ADDE - Getting Started.
   Type: **DSSERVE ADD MYDATA/IMAGES 1 9999 "ALL AREA FILES**
   Type: **IMGREMAP RTGOESS/FD MYDATA/IMAGES.4 STATION=KSEA PRO=MERC RES=5 SIZE=480 640 BAND=13**
9. List out the directory information for these images.
   Type: **IMGLIST RTGOESS/FD FORM=BAND;IMGLIST MYDATA/IMAGES.4 FORM=BAND**

```
Image file directory listing for:RTGOESS/FD Pos Satellite/ Date Time Center Res (km) Image_Size sensor Lat Lon Lat
Lon --- ------------- ------------ -------- ---- ---- ----- ----- ------------ 192 GOES-17 3 OCT 20277 00:40:31 0
137 Band: 1 0.47 um VIS aerosol-over-land 1.00 1.00 10848 x10848 Band: 2 0.64 um VIS clouds fog, insol, winds 0.50
0.50 21696 x21696 Band: 3 0.86 um Near IR veg/burn scar,aerosol, w 1.00 1.00 10848 x10848 Band: 4 1.37 um Near IR
cirrus cloud 2.00 2.00 5424 x 5424 Band: 5 1.6 um Near IR cloud phase, snow 1.00 1.00 10848 x10848 Band: 6 2.2 um
Near IR land/cloud, vege, snow 2.00 2.00 5424 x 5424 Band: 7 3.9 um IR Sfc, cloud, fog, fire, winds 2.00 2.00 5424 x
5424 Band: 8 6.2 um IR High-level WV, winds, rainfall 2.00 2.00 5424 x 5424 Band: 9 6.9 um IR Mid-level WV, winds,
rainfall 2.00 2.00 5424 x 5424 Band: 10 7.3 um IR Lower-level WV, winds & SO2 2.00 2.00 5424 x 5424 Band: 11 8.4 um
IR Total WV cloud phase, dust 2.00 2.00 5424 x 5424 Band: 12 9.6 um IR Total ozone,turbulence,wind 2.00 2.00 5424 x
5424 Band: 13 10.3 um IR Surface & cloud 2.00 2.00 5424 x 5424 Band: 14 11.2 um IR Imagery,SST,clouds,rainfall 2.00
2.00 5424 x 5424 Band: 15 12.3 um IR Total water, ash, and SST 2.00 2.00 5424 x 5424 Band: 16 13.3 um IR Air temp,
cloud hgt and amt 2.00 2.00 5424 x 5424 Image file directory listing for:MYDATA/IMAGES Pos Satellite/ Date Time
Center Res (km) Image_Size sensor Lat Lon Lat Lon --- ------------- ------------ -------- ---- ---- ----- -----
------------ 4 GOES-17 3 OCT 20277 00:30:31 47 122 Band: 13 10.3 um IR Surface & cloud 3.38 3.38 480 x 640
```

   Notice that the second image now has a different center lat/lon and is a different size than the original image.

10. Display the new image in frame 4 and draw a high-resolution United States map (the default).
    Type: **IMGDISP MYDATA/IMAGES.4 4 SF=YES;MAP FILE=OUTLUSAM**
11. Loop all of the GOES-West images.
    Press: **Alt L**
    By looking at these four images, you should be able to see how simple it is to create different products from the same source image (frame 1) without ever moving the source image from the remote server.

12. When finished viewing the loop, turn the looping off.
    Press: **Alt L**

Real-time Data Access - NWS Radar Imagery

Many of the same commands used to display satellite images can be used with National Weather Service (NWS) Radar data as well. In this excercise, you will access real-time NWS Radar imagery, find a NWS Radar station where precipitation exists (if any) and list and display that station's radar image.

The DSINFO and IMGLIST commands list information about the available images. Once located, you can display the images using the IMGDISP command. You can display images one at a time or in a sequence. An entire set of images can be loaded chronologically using one IMGDISP command.

1. Add the group RTNEXRAD into the client routing table and list all radar datasets from RTNEXRAD of type IMAGE.
   Type: **DATALOC ADD RTNEXRAD ADDE.UCAR.EDU**
   Type: **DSINFO IMAGE RTNEXRAD**

```
Dataset Names of Type: IMAGE in Group: RTNEXRAD Name NumPos Content ------------ ------
-------------------------------------- DAA 99999 Digital Accumulation Array DHR 99999 Digital Hybrid Reflectivity
DOD 99999 Digital One Hour Difference DPA 99999 Digital Precipitation Array DSD 99999 Digital Storm Total Difference
DSP 99999 Digital Storm Total Rainfall DTA 99999 Digital Storm Total Accumulation DVL 99999 Digital Vertical Liquid
H2O EET 99999 Enhanced Echo Tops HHC 99999 Hybrid Scan Hydrometeor Classification N0C 99999 Correlation Coefficient
Tilt 1 N0H 99999 Hydrometeor Classification Tilt 1 N0K 99999 Specific Differential Phase Tilt 1 N0Q 99999 Base
Reflectivity DR Tilt 1 N0R 99999 Base Reflectivity Tilt 1 N0S 99999 Storm-Rel Mean Vel Tilt 1 N0U 99999 Base
Velocity DV Tilt 1 N0V 99999 Radial Velocity Tilt 1 N0X 99999 Digital Differential Reflectivity Tilt 1 N0Z 99999 248
nm Base Reflectivity N1C 99999 Correlation Coefficient Tilt 3 N1H 99999 Hydrometeor Classification Tilt 3 N1K 99999
Specific Differential Phase Tilt 3 N1P 99999 1-hour Surface Rainfall N1Q 99999 Base Reflectivity DR Tilt 3 N1S 99999
Storm-Rel Mean Vel Tilt 2 N1U 99999 Base Velocity DV Tilt 3 N1V 99999 Radial Velocity Tilt 2 N1X 99999 Digital
Differential Reflectivity Tilt 3 N2C 99999 Correlation Coefficient Tilt 5 N2H 99999 Hydrometeor Classification Tilt
5 N2K 99999 Specific Differential Phase Tilt 5 N2Q 99999 Base Reflectivity DR Tilt 5 N2S 99999 Storm-Rel Mean Vel
Tilt 3 N2U 99999 Base Velocity DV Tilt 5 N2X 99999 Digital Differential Reflectivity Tilt 5 N3C 99999 Correlation
Coefficient Tilt 6 N3H 99999 Hydrometeor Classification Tilt 6 N3K 99999 Specific Differential Phase Tilt 6 N3Q
99999 Base Reflectivity DR Tilt 6 N3S 99999 Storm-Rel Mean Vel Tilt 4 N3U 99999 Base Velocity DV Tilt 6 N3X 99999
Digital Differential Reflectivity Tilt 6 NAC 99999 Correlation Coefficient Tilt 2 NAH 99999 Hydrometeor
Classification Tilt 2 NAK 99999 Specific Differential Phase Tilt 2 NAQ 99999 Base Reflectivity DR Tilt 2 NAU 99999
Base Velocity DV Tilt 2 NAX 99999 Digital Differential Reflectivity Tilt 2 NBC 99999 Correlation Coefficient Tilt 4
NBH 99999 Hydrometeor Classification Tilt 4 NBK 99999 Specific Differential Phase Tilt 4 NBQ 99999 Base Reflectivity
DR Tilt 4 NBU 99999 Base Velocity DV Tilt 4 NBX 99999 Digital Differential Reflectivity Tilt 4 NCR 99999 Composite
Reflectivity NET 99999 Echo Tops NTP 99999 Storm Total Rainfall NVL 99999 Vertical Liquid H2O OHA 99999 One Hour
Accumulation PTA 99999 Storm Total Accumulation
```

2. Erase frames 1 through 6.
   Type: **ERASE F 1 6**
3. Display a map of the United States on frame 1 and show the current weather conditions by displaying weather symbols.
   Type: **SF 1;MAP USA;SFCPLOT WXS**
   The parameter WXS displays weather symbols of the current weather conditions.

4. **Choose a station** in which rain, snow, or some other type of precipitation is occuring and **place the cursor** over the symbol.

5. Find any nearby NWS Radar stations using the STNLIST command.
   Type: **STNLIST TYPE=N**
   The TYPE keyword allows you to search for specific types of stations. In this example, you are only searching for NWS Radar stations (N) in the surrounding area.

   You should see an output similar to this:

```
Center Lat/Lon: 42:48:59 84:18:00 IDN ID Station Name Data Types ST CO DIST[KM] LAT LON ELE ----- -----
-------------------- ----------------- -- -- -------- ---------- --------- ---- --- DTX DETROIT N MI US 68.9
42:41:59 83:28:18 327 --- DTW Detroit TDWR N MI US 101.5 42:06:40 83:30:54 235 --- GRR GRAND RAPIDS NW MI US 101.8
42:53:38 85:32:41 237 Number of stations listed: 3
```

6. Choose a station ID and display the base reflectivity (tilt 1, 248 nmi range) image from that station (in this case, DTX, Detroit, Michigan) with a red map on frame 2.
   Type: **SF 2;IMGDISP RTNEXRAD/N0Q ID=DTX MAG=-2;MAP X 5**
7. Another way to display the same station is to use the predefined NWS Radar product aliases found in IMGDISP.CORE. For the *preprod* parameter IMGDISP uses the IMGDISP.* context files, where .* refers to .CORE, .USER, and .SITE. Display the IMGDISP.CORE context file.
   Type: **SEE IMGDISP.CORE NLINES=ALL**
   After the comments section of the IMGDISP.CORE file, you will see predefined products in sections like this:

```
# Products for WSR Base Reflectivity, tilt 1 (0.5deg) RADAR/WSR-BREF1 DATASET=RADAR/WSR-BREF1 EU=WSR-BREF REQUIRE=ID
WSR/BREF1 DATASET=WSR/BREF1 EU=WSR-BREF REQUIRE=ID MKX_BREF1 DATASET=RADAR/WSR-BREF1 EU=WSR-BREF ID=MKX MAG=-2 \
REFRESH='BAR (IMA) GRA=(GRA); \ EU MAKE 1 1 50 50 50 50 50 50 (IMA) (IMA); \ EU MAKE 2 2 WHITE WHITE (IMA) (IMA); \
EU MAKE 3 3 BLACK BLACK (IMA) (IMA); \ MAP X -8 DASH=1 X 4 COUNTY=ALL GRA=(GRA) IMA=(IMA); \ MAP X 7 GRA=(GRA)
IMA=(IMA); \ STNPLOT TYPE=FOUS COLOR=7 FONT=HELMO GRA=(GRA) NAV=(IMA); \ FRMLABEL LEV=2 3 IMA=(IMA) \ "(ID) WSR -
BASE REFLECTIVITY - TILT 1 AT (HHMM) ON (DAY)'
```

8. Take a closer look at each individual line of the MKX_BREF1 predefined product, to see exactly what it does.
   - **DATASET=RADAR/WSR-BREF1 EU=WSR-BREF ID=MKX MAG=-2 **
     The DATASET= keyword defines the ADDE dataset to use, EU= defines the enhancement file to use, ID= keyword defines the default station to display, and MAG= defines the magnification factor to apply when displaying the image. The "\" is the continuation line used

by the context file.
Note: The DATASET= keywords defined in IMGDISP.CORE use the Satellite Data Services/McIDAS-XCD naming convention. Users can define their own DATASET= in IMGDISP.SITE or IMGDISP.USER. Also, keywords can be overridden on the command line. We will do this with DATASET= in the following examples.

- **REFRESH='BAR (IMA) GRA=(GRA); **
  The REFRESH= keyword runs the remaining commands on each frame after the image is displayed. (IMA) and (GRA) are used to specify the current image and graphics frame. The BAR command creates and displays a grayscale bar, which turns into a color bar with the effects of the enhancements.
- **EU MAKE 1 1 50 50 50 50 50 50 (IMA) (IMA); **
  The three EU commands change the BREF image enhancement to have different background colors.
- **MAP X -8 DASH=1 X 4 COUNTY=ALL GRA=(GRA) IMA=(IMA); **
  The MAP command displays a map of gray dashed county lines using the current frame's navigation.
- **MAP X 7 GRA=(GRA) IMA=(IMA); **
  The second MAP command displays a high resolution map using the current frame's navigation.
- **STNPLOT TYPE=FOUS COLOR=7 FONT=HELMO GRA=(GRA) NAV=(IMA); **
  The STNPLOT command plots all FOUS14 stations in white with a font of HELMO, using the current frame's navigation.
- **FRMLABEL LEV=2 3 IMA=(IMA) **
  "(ID) WSR - BASE REFLECTIVITY TILT 1 AT (HHMM) ON (DAY)'
  The FRMLABEL adds a label to the frame, with the (ID), day (DAY), and time (HHMM) of the image.

9. Display the same station's three most recent NWS Radar images on frame 3-5, using the predefined product, MKX_BREF1, and the station you chose in step 6 (in this case, DTX). Use the RTNEXRAD dataset that was added above. If MKX is the station you chose, you do not need to specify ID=, since ID=MKX is the default.
   Type: **IMGDISP MKX_BREF1 ID=DTX DATASET=RTNEXRAD/N0Q ALL=3 5**
10. Loop the images to see the movement of the storm.
    Type: **LB 3 5**
    Press: **Alt L**
    Press: **Alt L**
11. Display the total surface rainfall accumulation from the RTNEXRAD dataset using the predefined PCPT (storm total precipitation) product on frame 6.
    Type: **SF 6;IMGDISP MKX_PCPT ID=DTX DATASET=RTNEXRAD/NTP**
12. Place the cursor over different areas of precipitation and use IMGPROBE to determine the total amount of precipitation over those selected ares.
    Type: **IMGPROBE**
    Press: **Alt G**
    Repeat this as many times as you like.

13. Exit IMGPROBE using Alt Q.
    Press: **Alt Q**

Real-time Data Access - Gridded Data

In this exercise, you will review the McIDAS grid commands by accessing real-time grids. If you need a review of grids and the grid file format, see the Basic Concepts section of the Grids and Grid Files Chapter.

1. List the grid files in the RTGRIDS dataset.
   Type: **DSINFO GRID RTGRIDS**
2. List all of the 850 mb relative humidity (RH) grids from today's 00 UTC NAM grid.
   Type: **GRDLIST RTGRIDS/NAM LEV=850 PAR=RH DAY=#Y TIME=00 NUM=ALL**
   All of the relative humidity grids available (usually from 0 to 60 hours) should be listed.

3. List the 12 hour forecasted 850 mb Relative Humidity grid point values between 37° and 39° N and 102° and 105° W from this dataset. Format the data as a real value with one decimal place.
   Type: **GRDINFO RTGRIDS/NAM LIST DAY=#Y TIME=0 LEV=850 FHOUR=0 PARAM=RH LAT=56 58 LON=132 135 FORMAT=F4.1**
   Row, column, latitude, longitude, and values are listed for each grid point falling in this geographical region (the southeast corner of Alaska). The amount of points listed depends on the resolution of the first matching grid:

```
Listing data from 12-hour forecast RH Grid at 850 MB from NAM Grid is at 0 UTC on 2021075 from dataset: RTGRIDS/NAM
Row Col Latitude Longitude Value Units Level Param 2 18 57.669 134.944 74.0 % 850 MB RH 3 18 57.091 134.620 65.0 %
850 MB RH 4 18 56.508 134.301 87.0 % 850 MB RH 2 19 57.842 133.865 98.0 % 850 MB RH 3 19 57.263 133.549 97.0 % 850
MB RH 4 19 56.680 133.238 90.0 % 850 MB RH 5 19 56.092 132.931 92.0 % 850 MB RH 3 20 57.431 132.473 92.0 % 850 MB RH
4 20 56.848 132.169 98.0 % 850 MB RH GRDINFO Done, Number of grids listed=1
```

4. Plot a map of the United States on Frame 6. Create a temperature advection grid using the U, V and T grids. Use the formula TADV = -(u*DDX(T) + v*DDY(T)). Label the new grid TADV and set the units to K/sec.
   Type: **SF 6;ERASE;MAP USA**
   Type: **GRDDISP RTGRIDS/GFS G1='PARAM T;LEV 850' G2='PARAM U;LEV 850' G3='PARAM V;LEV 850' MATH='-(G2*(DDX(G1))+G3*(DDY(G1)))' NEWPAR=TADV K/S NAV=C FHOUR=12 TIME=00 DAY=#Y**
   The G1, G2, ..., Gn keywords are used to define the grids you want to use. The MATH keyword defines the mathematical operation to perform on the grids specified with the *Gn* keywords. See the GRDDISP help section for more information on the use of these keywords.

5. Erase frames 1-4. Create a sequence of the 6, 12, 18, and 24 hour 500 mb height field today's 00 UTC UKMT model run. Note: Grid availability may change on the server.
   Type: **ERASE F 1 4**
   Type: **GRDDISP RTGRIDS/UKMT DAY=#Y TIME=00 FHOUR=6 12 18 24 LEV=500 PARAM=Z CINT=60 GRA=1 4 LSIZE=6 LINT=2 SF=YES**
6. Set the loop bounds to frames 1-4. Change the dwell rate so it pauses longer on the last frame. Then, start the loop.
   Type: **LS 1-4;DR 3*3 10**
   Press: **Alt L**
7. Stop the loop.
   Press: **Alt L**
8. Exit McIDAS.
   Type: **EXIT**

Graphical User Interface - GUI

Table of Contents:

Graphical User Interface (GUI)

In this last section of the Learning Guide, you will use the McIDAS Graphical User Interface (GUI) to display images and perform other actions previously done on the command line.

The examples listed in this chapter assume that you are using the same dataset names as those used at SSEC. If your dataset names are different, you may have to change the dataset names in some of the commands.

This lesson describes how to:

- set up your ADDE servers
- display and loop satellite images
- modify image colors
- modify graphic colors
- display and loop grid data
- copy images and grids
- display real-time surface and upper air data
- display weather watches and warnings
- configure the default map
- change cursor parameters
- create shortcut buttons
- search through the McIDAS command history
- schedule McIDAS commands

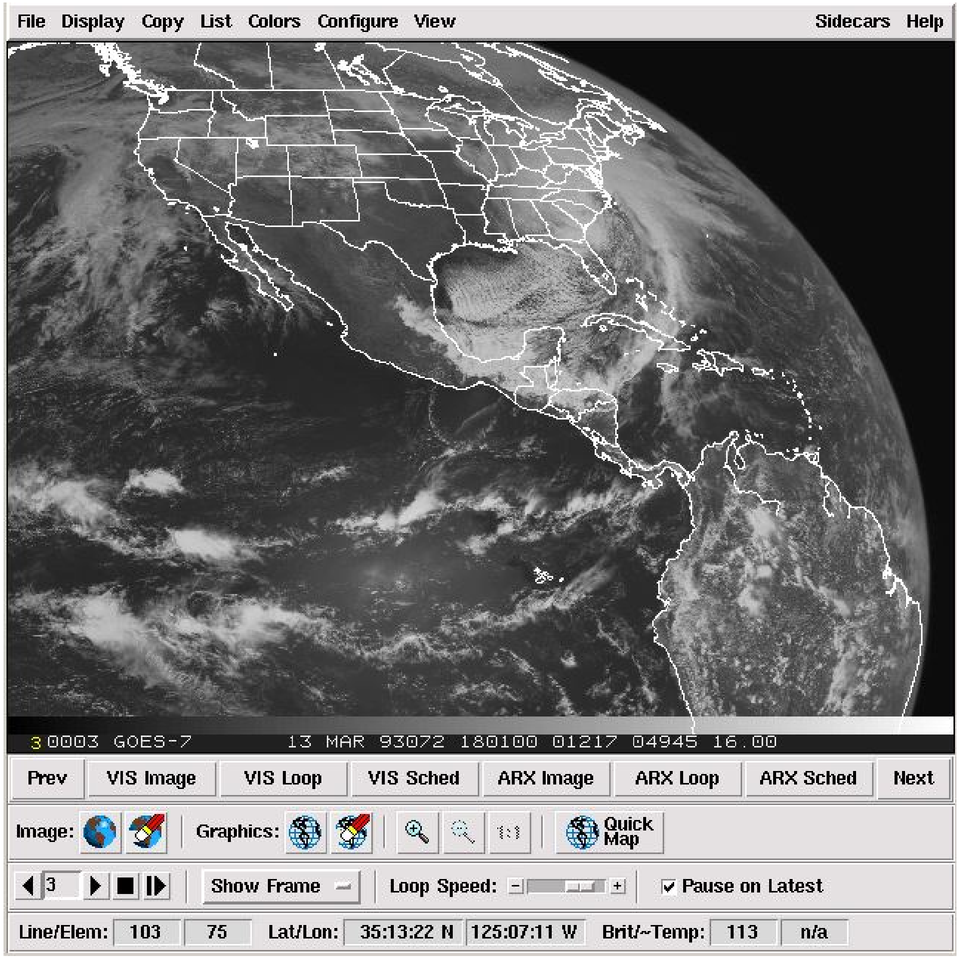
GUI - Basic Concepts

The McIDAS Graphical User Interface (GUI) lets you display images and data without using the McIDAS Text and Command Window. However, you can still run commands from the McIDAS Text and Command window while the GUI is displayed.

Starting and Exiting the GUI

There are two ways to start the McIDAS GUI. To start the GUI automatically when you start a McIDAS session, add the line **-c 'GUI'** to your **.mcidasrc** flie. To start the GUI when necessary, simply type GUI on the command line during a McIDAS session.

When you start the GUI, a menu bar and four toolbars are added to the McIDAS Image Window. The McIDAS Text and Command Window remains displayed. Once you have displayed an image in the GUI, the Image Window will look similar to this:



To exit the GUI, simply choose **Exit GUI** from the File menu. To exit both the GUI and the McIDAS session, select **Exit McIDAS** from the File menu.

Using the GUI

A single click of the left mouse button activates most menus and buttons. A double click is indicated when necessary. Buttons appear as labeled, raised areas. To move between fields in a dialog box, reposition the cursor and click, or press the Tab key. To move sliders, press the mouse button and drag the slider in the appropriate direction or use the arrow keys on the keyboard. Some dialog boxes have a **More** button. Click this button to display additional, but less used, options.

To control frame looping, use the buttons on the Frame Control Toolbar.

 To move backward one frame, click on the single left arrow button.

 To advance forward one frame, click on the single right arrow button.

 To start/pause the looping, press the play button.

To stop the looping, press the stop button.

Use the **Show Frame** button to select a frame to view. The frame number is displayed in the lower left corner of the Image Window. The current position in the loop is displayed above the slider. To control the **Loop Speed**, click on the + or **-** buttons or use the mouse to slide the scroll bar to the designated speed.

Use the **Pause on Latest Image** button to have the looping pause automatically on the lastest image of the loop.

To erase or toggle the Image and Graphic windows, use the buttons on the Image Control Toolbar.

To erase the graphics, click the erase graphics button. A right-click will allow you to select individual graphics colors to erase.

To turn the graphics off, click the graphics on button. A right-click will allow you to select individual graphics colors to toggle on/off.

To turn the graphics on, click the graphics off button.

To erase the image, click the erase image button.

To turn the image off, click the image on button.

To turn the image on, click the image off button.

The zoom buttons allow you to zoom the image and graphics in or out.

To zoom in, click the zoom in button.

To zoom out, click the zoom out button.

1:1 To reset the zoom to 1 to 1, click the zoom reset button.

Use the **Quick Map** button to display the default map with the current frame navigation.

The cursor location is indicated on the Cursor Status toolbar. The line/element location is always listed. If the frame is navigated, then the Latitude and Longitude of the cursor will appear. If an image is displayed, the Brightness value will be listed, and if the image is an infrared image, then the Brightness Temperature will also be listed.

The GUI Menus

- **File Menu**

  The File menu can be used to logon to McIDAS, update the ADDE servers, save frames in different image formats, open briefing windows, and exit McIDAS or the GUI.

- **Display Menu**

  The Display menu can be used to display images, grids, real-time surface and upper air data, weather watches and warnings, and maps.

- **Copy Menu**

  The Copy menu can be used to copy images and grids to local files.

- **List Menu**

  The List menu can be used to access weather text files, the workstation configuration, current frame information, and the command history.

- **Colors Menu**

  The Colors menu can be used to modify the graphics colors, the image colors and grayscale, and to do freehanded drawing.

- **Configure Menu**

The Configure Menu can be used to add frames, scheduler entries, change loop bounds, change the cursor, font, and colors, and to update the shortcut buttons. There is also an option to echo the GUI commands to the text window.

- **View Menu**

  The View Menu can be used to move the GUI toolbars to the top or the bottom, or to hide them.

- **Help Menu**

  The Help menu lists information about McIDAS, as well as some general information on some of the different buttons and features of the GUI.

GUI - Getting Started

In a previous lesson, you learned that the ADDE distributes data using networked servers and clients.

In this exercise, you will configure your GUI and use it to access all of the ADDE servers that you have added in the other sections of the Learning Guide.

1. Start a McIDAS session with the configuration GUI and confirm that McIDAS will start the GUI.
   At the Unix prompt:
   Type: **mcidas -config**
   Click: **Miscellaneous**
   Confirm that the "Start GUI upon McIDAS startup" checkbox is checked on.
   Click: **Start McIDAS**
2. A message may pop up explaining that your ADDE servers are not set up for the GUI. The ADDE servers for the GUI are separate from the command line, so it is necessary to retrieve them for the GUI.
   a. Click **OK** to begin to set up your ADDE servers.
   b. You can logon to McIDAS if you use SSEC SDS data servers. It is not necessary to logon for the Learning Guide exercises. Click **Continue** to continue the ADDE server retrieval.

   The rebuilding process can take several minutes to complete. Do not be concerned if it takes a while to build the list. You may see a message that states you are not logged on. This is normal, click **OK** to continue.

   Once the ADDE Server List shows the complete list of servers, the server table is complete. You can Click **Exit** and proceed to the next lesson.

3. If the ADDE Server Retrieval box did not pop up, update the ADDE servers.
   Select: **File/ADDE Servers**
   Click: **Rebuild ADDE List**
   You can logon to McIDAS if you use the SSEC SDS data servers. It is not necessary to logon for the Learning Guide exercises. Click **Continue** to continue the ADDE server retrieval.

   The rebuilding process can take several minutes to complete. Do not be concerned if it takes a while to build the list. You may see a message that states you are not logged on. This is normal, click **OK** to continue.

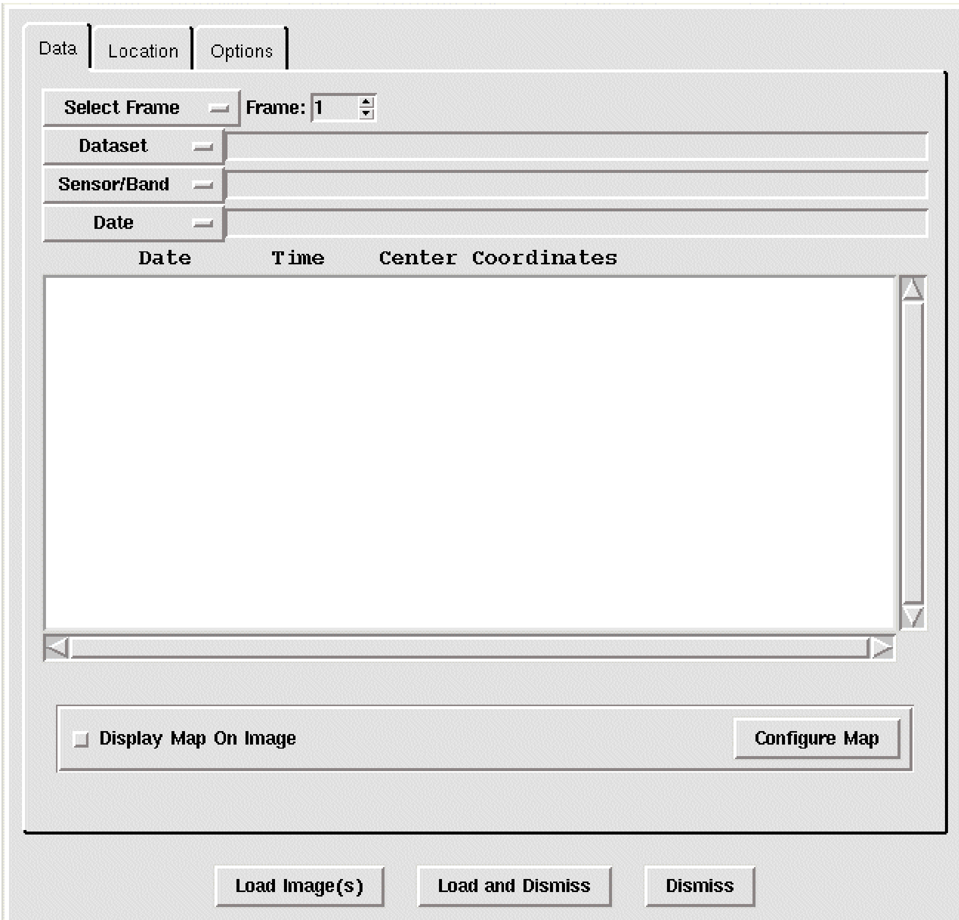
   Once the ADDE Server List shows the complete list of servers, the server table is complete. You can Click **Exit** and proceed to the next lesson.

GUI - Displaying a Satellite Image

In this exercise, you will display a satellite image using the GUI.

1. Select the Image GUI from the **Display** Menu.
   Select: **Display/Image**

   The Image Select GUI should come up and look similar to this:

2. There are four menu buttons that are used to select an image: **Select Frame**, **Dataset**, **Sensor/Band**, and **Date**. Display the Band 13 Full Disk image from GOES East and display the image on frame 2.
   Select: **Select Frame/ALL FRAMES/2**
   Select: **Dataset/RTGOESR/FDC13**
   Highlight the most recent image in the list by clicking on its listing in the white window.
   Select: **Load Image(s)**
3. Display the same image on frame 3, but include a map and display only part of the data.
   Select: **Select Frame/ALL FRAMES/3**
   Click: **Display Map On Image**
   Click: **Location**
   The Location tab contains an image of a globe. After selecting an image, the area of coverage of the image will be displayed on the globe as a yellow box. Once the bounds are displayed, you can alter the coverage shown by using the left mouse button to drag the selection bounds, and the middle mouse button to set the latitude and longitude center point, or by filling in the lat/lon or station ID, and the Line/Ele magnification.
   Use the mouse to select a new data coverage area.
   Click: **Load Image(s)**
   This will load the image, but not exit the Image GUI. Compare the image with the coverage shown on the **Location** tab.

4. Close the Image Select GUI window.
   Click: **Dismiss**

GUI - Looping Satellite Images

In this exercise, you will create loop bounds, and display a loop of images.

Creating Loop Bounds

To loop Satellite Images, we must first create a set of loop bounds to use when displaying the images. Create a loop bound using frames 1 through 3 and name it 'Satellites.'

1. Bring up the **Loop Bounds** GUI under the **Configure** menu.
   Select: **Configure/Loop Bounds**
2. Start a new loop.
   Click: **New Loop**
3. Select your loop bounds by clicking on the first frame, and dragging the mouse over the desired frames to be included in the loop bounds.
   Select: **Frames 1 through 3**
4. Rename 'Loop 1' to 'Satellites.'
   In the Loop Name field, change 'Loop 1' to 'Satellites'.
5. Save the Loop.
   Click: **Set Loop**
6. Dismiss the Loop Manager GUI.
   Click: **Dismiss**

Displaying a Loop of Satellite Images

1. Next, you will display the 3 latest GOES-West, band 1 images in the 'Satellites' loop. (Choose visible imagery during the day.)

2. Use the Image Select GUI to choose the three most recent GOES-West CONUS satellite images.
   Select: **Display/Image**
   Select: **Dataset/RTGOESS/CONUSC01**
   Select 3 images by clicking on the first and dragging the mouse button over the rest.
3. Start the loop on frame 1, display a map on each image, select the geographical coverage of the images, and display the loop.
   Select: **Select Frame/Satellites/1**
   Click: **Display Map on Image**
   Click: **Location**
   Use the mouse to select a new data coverage area.
   Click: **Load and Dismiss**
   This will load all selected images and dismiss the GUI when the loading process is complete.

4. Start the looping of the images.
   Click: **Play**
   Notice how the looping follows the 'Satellite' loop bounds and only loops the first three frames. You may change the loop speed by sliding the Loop Speed slider or clicking on the plus and minus buttons. Moving the slider to the right decreases the dwell rate, thus increasing the looping speed. Moving the slider to the left decreases the looping speed.

5. Make the loop pause on the most recent image.
   Click: **Pause on Latest Image**
   Notice how the loop now pauses longer on frame number 3 than it does on frames 1 and 2.

6. Stop the looping.
   Click: **Stop**

Displaying a Loop of Radar Images

1. Next, you will create a loop named RADAR and display a loop of real-time WSR radar images from a station that you select.

2. Create a new loop called 'RADAR'.
   Select: **Configure/Loop Bounds**
   Click: **New Loop**
   Select six frames, by clicking frame 1 and dragging the mouse down to frame 6
   Rename 'Loop2' to 'RADAR'
   Click: **Set Loop**
   Click: **Dismiss**
3. Display a loop of WSR radar images.
   Select: **Display/Image**
   Select: **Select Frame/RADAR/1**
   Select: **Dataset/RTNEXRAD/N0Q**
   Click: **State**
   Click: ***the state of your choice***

Click: **Station**
Click: ***the station of your choice***
Select the six most recent images by clicking the first image and dragging the mouse down over the next five images.
Click: Confirm the box to Use Standard Radar Settings is checked.
Click: **Load and Dismiss**

4. Start the looping of the images.
   Click: **Play** 
5. Stop the looping.
   Click: **Stop** 
6. When you are done viewing the RADAR loop, erase the images and graphics from the current frame.
   Click: **Erase Image** 
   Click: **Erase Graphics** 

GUI - Modifying Image and Graphics Colors

Modifying Image Colors

In this exercise, you will display an infrared image from a GOES-East dataset, restore the default enhancement, and modify the image and graphics colors.

1. Display a GOES-East image and map on the current frame, using data from band 13 of the RTGOESR/FDC13 dataset.
   In the Display/Image GUI, select: the **RTGOESR/FDC13 dataset**, and to **display a map on the image**.
   Select one image from the image list, load the image, and dismiss the Image Select window.
2. Obviously, these colors are not the colors of a normal infrared image. Even though the frames were erased, the radar enhancement is still assigned to that frame. So, you'll need to restore the correct colors by modifying the Image Colors.
   Select: **Colors/Change Image Colors**
   Click: the **Reset** tab
   The default option in the Reset tab is to reset the current frame only.
   Click: **Reset**
3. Activate the GUI where you'll create your own enhancement.
   Click: **Make**
   The **Make** tab of the Image Colors GUI contains four options in which you can change or create image colors: 'Using Color Names,' 'Using Color Intensities,' 'Using ENH,' and 'Using EB.'
   - The 'Using Color Names' button consists of two scroll bars labeled 'Begin' and 'End.' The scroll bars allow you to select a beginning and ending color from one of the 35 predefined McIDAS colors.
   - The 'Using Color Intensities' button also has the two scroll bars, but allows you to choose the exact amount of red, green, and blue intensities to create your own colors.
   - The 'Using ENH' button is a graphical way to change the intensities of red, green, and blue.
   - The 'Using EB' button starts the EB command which invokes the interactive mouse-controlled contrast stretching. Press the right mouse button to end the process, leaving the current enhancement.

   Start out by using the predefined McIDAS colors.
   Click: **Using Color Names**
4. Use the predefined color 'Blue' to begin at a grayscale of 145, and 'Goldenrod' to end at a grayscale of 210.
   Slide: the **Begin** scrollbar to 145
   Slide: the **End** scrollbar to 210
   Select: **Beginning Color/Blue**
   Select: **Ending Color/Goldenrod**
   Click: **Apply Enhancement**

   Look at the Image window. You should see that all brightness values from 145 to 210 are colored from Blue to Goldenrod, depending on the specific value. All other brightness values will remain in their original grayscale form.

5. Repeat Step 4 several times, changing the colors and input values each time.

6. Once you have finished altering the image colors, dismiss the color selection GUI and save the enhancement as LEARNING.ET.
   Click: **Dismiss**
   Click: **Save**
   Type: **LEARNING**
   Click: **Save**
7. Reset the original image enhancement and restore LEARNING.ET to the image once again.
   Click: **Restore**
   Select: **Select Enhancement File/LEARNING.ET**
   Click: **Restore**
8. Restore the original image enhancement.
   Click: **Reset**
   Select: **Frame Selection/Set for Current Loop**
   Click: **Reset**
   Click: **Dismiss**

Modifying Graphics Colors

1. Change the color of the map in the current frame.
   Select: **Colors/Change Graphics Colors**
   Click: **Create a New Color**
   Slide: **the Red, Green, and Blue Intensity bars** until you've created a color you like
   Click: **Apply Enhancement**
   Click: **Dismiss**
   Look at the image window, and you will notice the changes to graphic color 1 took effect immediately.

2. Save the new graphics table as LEARNING.GRX.
   Click: **Save**

Type: **LEARNING** in the File Name field
Click: **Save**

3. Restore the default graphics colors.
Click: **Reset**
Select: **Frame Selection/Current Frame Only**
Click: **Reset**

4. Close the Graphics Colors window.
Click: **Dismiss**

GUI - Plotting and Looping Grids

In this exercise, you will display and compare grids from two different models, and then display a loop of grids.

Displaying Grid Data over an Image

1. Display the previous day's 48 hour 700 mb relative humidity forecast from the 12:00 UTC GFS model run in green over the current GOES-East image.
   Select: **Display/Grid**
   Select: **Dataset/RTGRIDS/GFS**
   If the RTGRIDS/GFS dataset contains too many grids to sort, a day and time modification box will pop up for you to narrow the search and try again.
   Enter: *yesterday's Julian day* - if needed in modification box or in **Date** field
   Select: **Time/12** - if needed in modification box
   Click: **OK** - if needed in modification box
   Select: **Parameter/RH**
   Select: **Level/700 MB**
   Click: the 48 hour forecast (Fhour = 048)
   Click: the **Options** tab
   Select: **Contour Color/Green**
   Click: **Load Grid(s)**
2. Load the same or similar data from the NAM model run in a different color and compare the results between the two models.
   Select: **Contour Color**/a different color of your choice
   Click: the **Data** tab
   Select: **Dataset/RTGRIDS/NAM**
   Select: **Parameter/RH**
   Select: **Level/700 MB**
   Select: **Date/*yesterday***
   Select: the 12:00 UTC 48 hour forecast
   Click: **Load and Dismiss**

Displaying and Looping Grid Data on a map

1. Erase the images and graphics on frames 1-6.
   Click: **Erase Image**
   Click: **Erase Graphics**
   Click: **Advance Frame**
   Repeat process until all frames have been erased.
2. Select a MSL pressure forecast from yesterday's 12Z NAM model run to be displayed on frame 5.
   Select: **Display/Grid**
   Select: **Select Frame/ALL FRAMES/5**
   Select: **Dataset/RTGRIDS/NAM**
   Select: **Projection/LAMB**
   Select: **Parameter/P**
   Select: **Level/MSL**
   Select: **Date/*yesterday***
   Click: a forecast grid to display from the list
3. Display the grid over the United States.
   Click: the **Options** tab
   Select: **Map Boundaries/USA**
   Click: **Load Grid(s)**
4. Display the same grid, but this time as part of a loop in frames 1-3.
   Click: the **Data** tab
   Select: **Select Frame/Satellites/1**
   Select: choose three grids in the list with different Fhours by clicking the first grid and holding the ctrl button down while clicking the next two grids chosen
   Click: **Load and Dismiss**
5. Loop the grids to see the movement of any weather systems.
   Click: **Play**
6. When done viewing the grids, stop the loop.
   Click: **Stop**
7. Erase the images and graphics on frames 1-6.
   Select: **Show Frame/ALL FRAMES/1**
   Click: **Erase Graphics**
   Click: **Advance Frame**
   Repeat process until all frames have been erased.

GUI - Copying Images and Grids

Now that you are familiar with loading and displaying images and grids, you will learn how to copy images and grids.

Copying Images

The image copy GUI has four different tabs which you can use to specify different aspects of the copy command.

- The Image Selection tab is used to select the copy method and choose the source and destination datasets.
  - **Copy Single Image** allows you to choose the specific image to copy.
  - **Copy Latest Image** automatically selects and copies the lastest image from the dataset.
  - **Update Dataset** copies the latest image from the dataset after it rechecks the dataset for the most recent image.
  - **Copy Entire Dataset** copies the entire source dataset to the destination dataset.
- The Band tab allows you to choose which band or bands you wish to copy.
- The Location tab allows you to specify the center point of the image as either the upper left hand corner of the image, a latitude and longitude point, or a specific station.
- The Size and Magnification tab lets you change the Line/Element magnification controls, as well as choose the image size to be copied.

1. Create a local dataset MYDATA/TESTING using position numbers 51 to 60.
   Select: **File/ADDE Servers**
   Double Click: **MYDATA**
   Click: **Add Local Entry**
   Enter: **TESTING** under Descriptor, **Test Images** under Description, **51** under Start Position, and **60** under End Position
   Click: **OK** and exit both ADDE GUI windows.
2. Copy a GOES-West full disk image to the local directory MYDATA/TESTING.
   Select: **Copy/Image**
   Click: **Copy Single Image**
   Select: **Source Dataset/RTGOESS/FDC13/OK**
   Select: *the second image in the list* (note its day and time)/**OK**
   Select: **Destination Dataset/MYDATA/TESTING/OK**
   In the Band tab, click: **Band 13**
   In the Size and Magnification tab, slide: *Line Mag = -6* and *Elem Mag = -8*
   Click: **Copy** and **Dismiss**
3. Next, display the image you just copied on frame 2. Display a white map over the image.
   Select: **Display/Image**
   Select: **Select Frame/ALL FRAMES/2**
   Select: **Dataset/MYDATA/TESTING**
   Click: *Date and time of image copied in step 2 above*
   Click: **Display Map on Image**
   Click: **Configure Map**
   Select: **Map Color/White**
   Click: **Select**
   Click: **Load and Dismiss**

GUI - Copying Images and Grids (Continued)

Copying Grids

The grid copy GUI has fewer options than the image copy GUI, but it still allows you the significant ones, like choosing specific beginning and ending grid numbers, along with the option to delete the destination set you will be copying the grids to.

In this lesson, you will use the local grids copied from this grid lesson. This test dataset will be used because the GRDCOPY GUI does not support GRIB data served from a database, which is how McIDAS-XCD stores real-time grids. Although no real-time data will be used in this exercise, you will still learn the basics of copying grids with the grid copy GUI.

1. If you did not complete the grid lesson mentioned above, run these commands from the command line to create a grid dataset and GRDCOPY grids into it to use in the following GUI exercise.
   Type: **DSSERVE ADD MYDATA/TEST-GRIDS GRID 4000 4009 "Test grids**
   Type: **GRDCOPY RTGRIDS/GFS MYDATA/TEST-GRIDS.1 DEL=YES DAY=#Y TIME=0 FHOUR=24 NUM=20**
2. Open the Grid Copy Window.
   Select: **Copy/Grids**
3. Select your Source Dataset and Grid to be copied.
   Select: **Source Dataset/MYDATA/TEST-GRIDS/OK**
   Click: *Choose Dataset Position 1*
   Click: **OK**
   Click: **Select Source Grid(s)**
   Select: **Sort Grids by:/Parameter**
   Click: *Choose a RH grid for example*
   Click: **OK**
4. Select the Destination Dataset.
   Select: **Destination Dataset/MYDATA/TEST-GRIDS/OK**
   Select: **Browse Contents of Destination Dataset**
   Click: *a grid file to copy over other than Dataset Position 1*/**OK**
   Click: *Delete Destination Grid(s)*
   Click: **Copy**
   Click: **Dismiss**

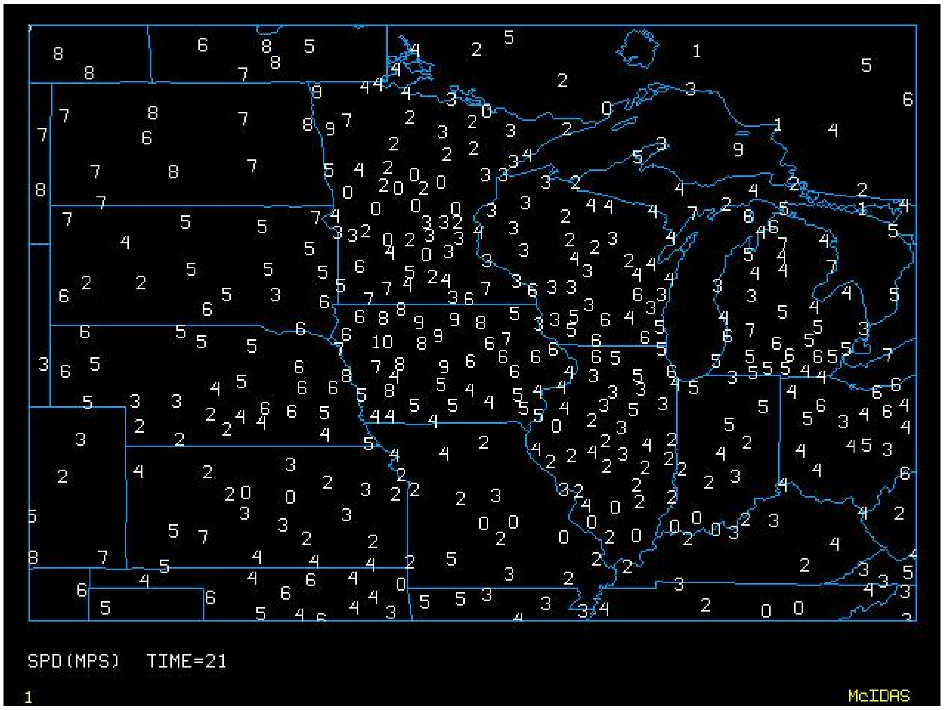
GUI - Plotting Surface and Upper Air Data

In this exercise, you will plot real-time surface and upper air observations (Please note that these two GUIs currently only work for real-time data, even though all MD datasets are listed under the dataset menu button).

Plotting Surface Data

1. Switch back to looping all frames, and erase the image and graphics in all six frames.
   Select: **Show Frame/ALL FRAMES/1**
   Click: **Erase Image** 
   Click: **Erase Graphics** 
   Click: **Advance Frame** 
   Repeat process until all six frames (1-6) have been erased.
2. Display the current wind speeds on frame 1. Display the data in meters per second over the Midwest using the dataset RTPTSRC/SFCHOURLY. Display a sky blue map with white label colors.
   Select: **Display/Real-time Surface Data**
   Select: **Select Frame/ALL FRAMES/1**
   Select: **Dataset/RTPTSRC/SFCHOURLY, Parameter/Wind Speed, Units/Wind/MPS, and Time/*Current Time***
   Click: ***Today's Date***
   Click: **Options** Tab
   Select: **Map Boundaries/MID**
   Select: **Map Color/SKY**
   Select: **Label Color/WHITE**
   Click: **Load Point(s)**

   The plot should look similar to this:



3. Now, look at the map and find the state with the highest wind speed.

   Use that state when defining the map to display the wind gusts on frame 2.
   Click: the **Data** tab
   Select: **Select Frame/ALL FRAMES/2**
   Select: **Parameter/Wind Gusts**
   Click: the **Options** tab
   Select: **Map Boundaries/*the state you chose***
   Click: **Load and Dismiss**

Plotting Upper Air Data

1. Erase the graphics from the first two frames using the method in step 1 above, and bring up the DISPLAY/Real-time Upper Air Data GUI.
2. Using the method described above, display today's 0 UTC 500 mb temperature from the RTPTSRC/UPPERMAND dataset over the United

States.

3. Dismiss the Real-time Upper Air GUI.

GUI - Plotting Weather Watches and Warnings

The Weather Watch GUI displays severe thunderstorm watches and warnings, tornado watches and warnings, winter storm watches and warnings, snow advisories, blizzard warnings, and flood watches and warnings. The GUI allows you to easily select which of these to plot, whether you want the watches plotted as watch boxes or county outlines, and whether you want the areas filled with a solid color, a transparent color, or just outlined. You can also select which map to use for the display and what time period to display.

In this exercise, you will plot the weather watches and warnings for the current time period.

1. Plot the watches and warnings in the United States within the last three hours on frame 3.
   Select: **Display/Weather Watch**
   The default of the weather watch GUI is to display all watches and warnings on a map of the United States, so for this example you will not need to modify anything on the Event Types tab except the frame selection, and nothing on the Map Settings tab.
   Select: **Show Frame/ALL FRAMES/3**
   Click: **Day and Time**
   Select: **Beginning Time/*three hours ago***
   Select: **Display and Dismiss**
2. Display an IR image over the United States on frame 4.
   Select: **Display/Image**
   Select: **Select Frame/ALL FRAMES/4**
   Select: **Dataset/RTGOESR/CONUSC13**
   Click: the most recent image
   Click: **Load and Dismiss**
3. Plot your choice of watches and warnings on the image.
   Select: **Display/Weather Watch**
   Select: which watches and warnings you'd like to plot
   Click: **Map Settings**
   Click: **Use Existing Frame Navigation**
   Click: **Display and Dismiss**

GUI - Configuring Your GUI

Some of the things that you do in the GUI are things that you want to work the same way every time that you run the GUI. For instance, in an earlier lesson, you set up the ADDE servers for the GUI. Servers do not change that often, so it's not something that you'll want to have to do every time you start the GUI. The server list is saved and will not have to be re-created every time you run the GUI. Other things that you can configure once and are automatically saved are your map preferences, your cursor preferences, and your short-cut buttons.

In this exercise, you will configure your GUI maps, cursor, and short-cut buttons.

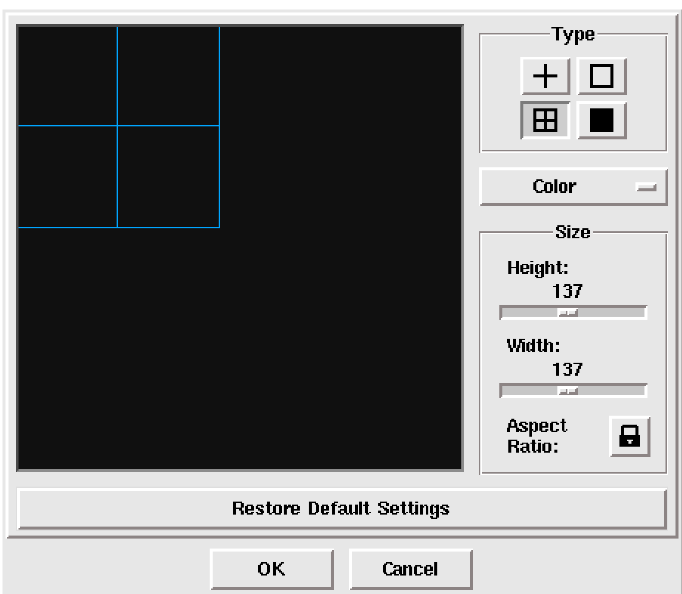
Configuring Your Map Preferences

1. Configure your map by selecting your map color preference. It is recommended to leave the Map Name as "Default Map", so that the command GUI that you are using can choose the best map for your application.
   Select: **Display/Map**
   Select: **Map Color/*choose your favorite color***
2. Select your preferences for the latitude/longitude lines on your maps. If you do not want lat/lon lines on your maps, do not check the "Draw LAT/LON Lines" box.
   Click: **Draw LAT/LON Lines** (if desired)
   Select: the color and contour interval for the lat/lon lines
3. Check your new map settings, and when you are satisfied with them, dismiss the map configuration GUI.
   Click: **Draw Map** to check your settings
   Click: **Draw & Dismiss**

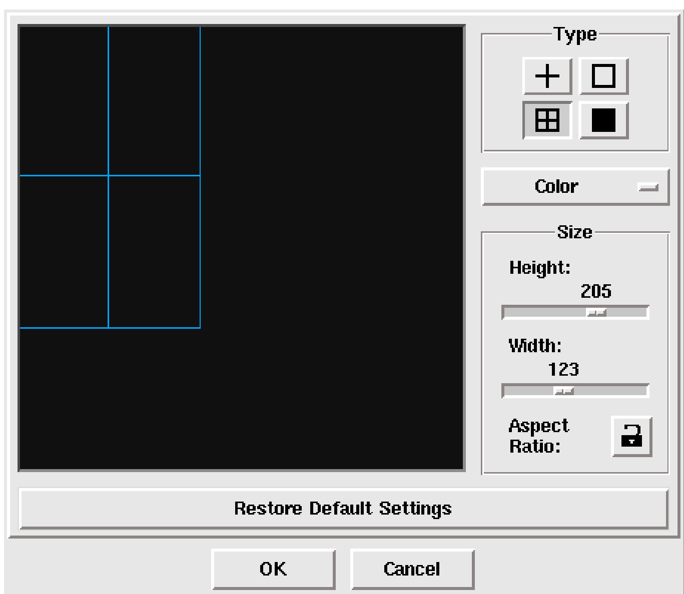
Changing Your Cursor

In McIDAS, there are four different cursor shape options: crosshairs (default), a box, a box with crosshairs, and a solid box. With the GUI, you are able to select the shape option and color, and then manually make the cursor smaller or bigger by clicking on the image and dragging the cursor. You can also change the size by moving the size scales for either the height or width. Directly below the size scales, you should see a button with a picture of a lock on it. Clicking on this button will allow you to 'unlock' the aspect ratio of the square. This means that instead of just different sized squares, you can make a rectangular shape with the cursor. The 'Restore Default Settings' button will restore the cursor to a 31 x 31 red crosshair cursor. Look below to see the difference between the locked and unlocked GUI.

Locked



Unlocked



1. Change the shape of your cursor.
   Select: **Configure/Cursor Definition**
   Change the size, shape, and color of your cursor, until you have a cursor that you like.
2. Save or cancel the changes to your cursor.
   Click: **OK** to make this your new cursor, or
   Click: **Cancel** to keep the default cursor

Creating Shortcut Buttons

Shortcut buttons are located on the User-Defined Button Toolbar. They can be used to run specific commands, or a set of commands that you frequently use, without having to type in the command itself. The shortcut buttons can also be made into drop-down menus, allowing the user to create several buttons of the same type, and file them under one listing for easy access and organization.

1. Create a shortcut button named 'temp' which erases the frame and displays the temperatures across the United States.
   Select: **Configure/Shortcut Buttons**
   Click: **New Button**
   Under Label, type: **temp**
   Under Popup Text, type: **US Temps**
   Under Commmand, type: **erase f;sfcplot t usa**

   When entering commands, they can be entered just as you would do with the text window, including the use of the semicolon. Also, your commands can be entered as lowercase. When checked, the checkbox at the bottom labeled "Force command string to upper case" will automatically convert the text to uppercase, just as they are in the McIDAS Text and Command window.

2. Click **OK** on both editors. The second **OK** will also dismiss the GUI, because it is necessary in order for the changes to take effect. You should now see your command 'temp' on the first shortcut button. It should look like this:

3. Click on the 'temp' shortcut button to make sure it works. You should see a map of the United States with the current temperatures displayed.

4. Create a drop-down menu button to display the latest RTNEXRAD/N0Q radar image over Buffalo, New York. Plot the current temperature contours over the image and name the command 'Buffalo'. Label the drop-down menu 'Radar'.
   Display the shortcut button GUI and set it up to create a new drop-down menu button.
   Select the Button Type: **Drop-down menu**
   Under Label, type: **Radar**
   For the Popup Text, type: **US Radar Imagery**
   Click: **Edit Menu Items**
   Click: **Add New Menu Item**
   Under Label, type: **Buffalo**
   Under Command, type: **EG;IMGDISP RTN_N0Q ID=BUF;SFCCON T**
5. Save the new button and try it out. It should work exactly like the text being entered into the text window.
   Select: **Radar/Buffalo**

   You can continue to create as many shortcut buttons as you like. You can also highlight and edit existing buttons, as well as create additional menu items under the 'Radar' menu you just created. The **Up** and **Down** buttons can be used to change the order in which the buttons are displayed. When the first row of buttons is full, use the **Next** and **Prev** buttons on the GUI to switch to the next set.

GUI - Recalling McIDAS Commands

Using the Command History GUI

The Command History GUI shows all McIDAS commands that have been entered, both on the command line or through the GUI. With the McIDAS Command History GUI, you can search previously entered commands, edit and execute them again.

There are three major sections in the McIDAS Command History Interface: the list of commands in the McIDAS Command History, the Search Command History controls, and the Execute McIDAS Command section where you can enter and execute commands.

The number of commands in the McIDAS Command History is equal to the value set in the *-ih* flag in the **$HOME/.mcidasrc** file. All commands are automatically saved in the **MCCMDHIST.TXT** file located in the **$HOME/mcidas/data** directory.

1. Search the Command History GUI for the Buffalo, NY RTNEXRAD radar image you displayed with the IMGDISP command in the previous exercise.
   Select: **List/Command History**
   Under "Search for:" enter: IMGDISP
   Click: **Find**
   Notice each time **Find** is clicked, each instance of IMGDISP is highlighted and entered in the "Execute McIDAS Command" field. Continue to click **Find** until the Buffalo command is found.

2. Edit the Buffalo radar IMGDISP command in the "Execute McIDAS Command" entry window. Erase the graphics, and display the first most recent RTNEXRAD radar image over Tulsa, Oklahoma.
   Change "RTN_N0Q" to "RTN_N0Q.-1"
   Change "ID=BUF" to "ID=INX"
   Prepend an "EG;" to the beginning of the command
   Click: **Go**

3. Practice searching the command history using different search options. When you are done, dismiss the Command History GUI.

   To dismiss the GUI:
   Click: **Dismiss**

GUI - Scheduling McIDAS Commands

With the McIDAS Scheduler commands (SKE, SKL, SKU, and SKED), you can set up McIDAS to automatically run commands at certain times of the day and at certain intervals. You can display images and graphics automatically, or you can list out data to the text frame, or you can even save images as GIF or JPEG images automatically. The Scheduler GUI allows you to easily add, delete, and edit scheduled commands.

In this exercise, you will create your own scheduler entries using the Scheduler GUI.

1. Open the Scheduler GUI, and confirm that the scheduler is turned ON.
   Select: **Configure/Scheduler**
   If you have any scheduler processes running, you will notice they are listed in the Main window. If you do not have any scheduler processes running, there will be a message saying "The Scheduler file, SKEDFILE is empty".
   Click: **Status**
   The light bulb should be lit (yellow), indicating that the scheduler is ON. If it's not ON, click the **Status** button again, until it is lit. If it asks you to start a scheduler process, click **"Yes"**.

2. Add a scheduler entry that will display the current surface observations over Seattle, WA for the next two hours, starting at the upcoming hour (##:00:00).
   Click: **Add**
   Under "McIDAS Command", type: **SFCLIST KSEA**
   Select: **Starting Time** / *the next upcoming top of the hour*
   Select: **Repeat / 2**
   Click: **Save Entry**

   This entry runs the SFCLIST command for the next two hours starting with the upcoming hour.

   If you choose to repeat the command "MANY" times, the command will run at the specified time interval (default of once every hour) for as long as you keep your scheduler file (**SKEDFILE**).

3. Execute the SFCLIST scheduler entry and dismiss the Scheduler GUI.
   Click: *the SFCLIST scheduler entry you just entered*
   Click: **Execute**
   The SFCLIST command is run and the output is displayed in the text command window. Executing the scheduler entry does not decrement the counter or change the scheduled run time.

4. When you are finished, close the Scheduler GUI and exit McIDAS.
   Click: **Dismiss**
   Select: **File/Exit McIDAS**
   Click: **Yes**
