# 7. Overlay Utility

IRIS has a flexible overlay feature for drawing overlays, or maps displayed on top of other IRIS/Open products. Overlays are used for product output and the real-time display. The overlays used in product output are specified in the Overlay menu.

An overlay can consist of the following:

- Geographical and political boundaries displayed with or without latitude and longitude lines or range rings.
- Text strings to label areas of interest, such as cities.
- Bitmap Icons that can be constructed by the user and applied to the overlay. Icons can represent any feature, for example, airports, train stations, etc.
- The lines, text and icons mentioned above can be separated into different layers within the overlay file. Then at display time, either all or only a subset of these layers may be displayed giving yielding overlays that appear different based on which layers are active. Layers can be drawn in different colors.
- Underlays are filled regions of color displayed under the radar images where there is no weather data. Underlays are typically used to indicate areas of water.

Each overlay is defined in an ASCII file using a connect-the-dots approach (sometimes called a vector approach). This allows overlays to be drawn to any scale factor.

The **overlay** utility is an interactive tool to create and modify your own overlays. It works in a window on your workstation. Typically, you get the basic map (coastlines, borders, rivers) from SIGMET and fine tune it according to your local needs. It's often convenient to make separate layers for different interests (river catchments, airports). IRIS supports a maximum of 20 overlay files.

IRIS overlays are ASCII files so you can edit them with any text editor (vi, or emacs which is distributed with IRIS).

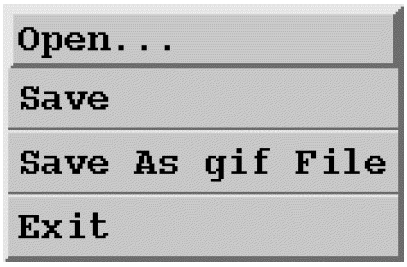
### In this chapter:

| | |
|---|---|
| ***Invoking overlay*** | **Section 7.1** |
| ***Listing and Printing Overlay Files*** | **Section 7.2** |
| ***Viewing an Overlay with overlay*** | **Section 7.3** |
| ***Format of Overlay Files*** | **Section 7.4** |
| ***Format of Catchment Files*** | **Section 7.5** |
| ***Creating and Editing Overlay Files*** | **Section 7.6** |

## 7.1 Invoking Overlay

### command: overlay

Overlay starts as a default with a empty window on your workstation and two menus: File and Options. Settings from your previous session are still valid when you restart. The File menu is shown in the figure below.




Your first action is to open a .ovr file. Overlay files in use on are stored on disk in a directory called /usr/sigmet/config/overlay. The Product Output process always looks for overlay files in that directory.


Then you want to see some options.

The first box lets you see the latitude and longitude of your cursor. You can move your cursor around your map with the mouse, but when you want to find a precise point it's often easier to use the arrow buttons of the keyboard which let you move pixel by pixel.

On the second line, you can select the mode to show latitude and longitude: either as degrees and decimals or degrees, minutes and their decimals.

From the third line, you get a menu shown below


In the first box you define the horizontal size of the map in km. Two next ones let you move around your original map and create subareas. Layers to draw defines if you want to work with all of your map layers or only a subset.

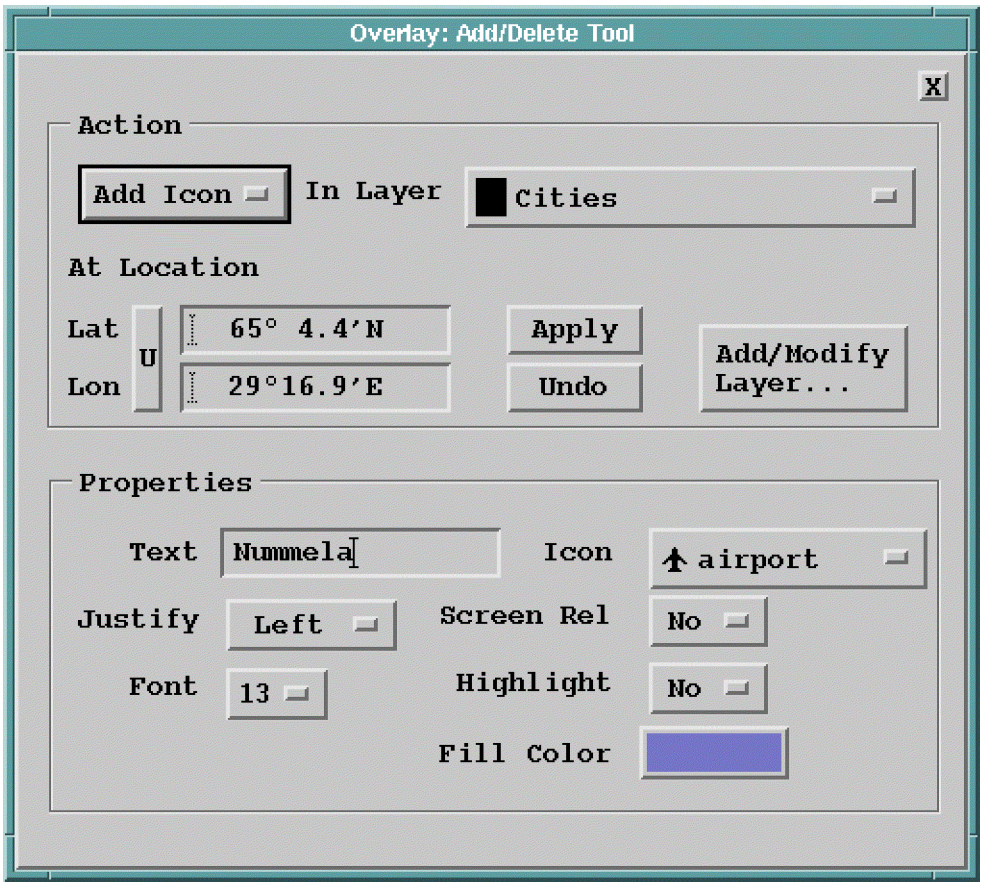
In the lower part, you can tick six options to be shown:

- Fill points indicate the areas, typically lakes, you want to be colored with a certain color.
- Lat Lon gives a latitude–longitude grid
- Range Rings are centered at a predefined point, typically a radar site

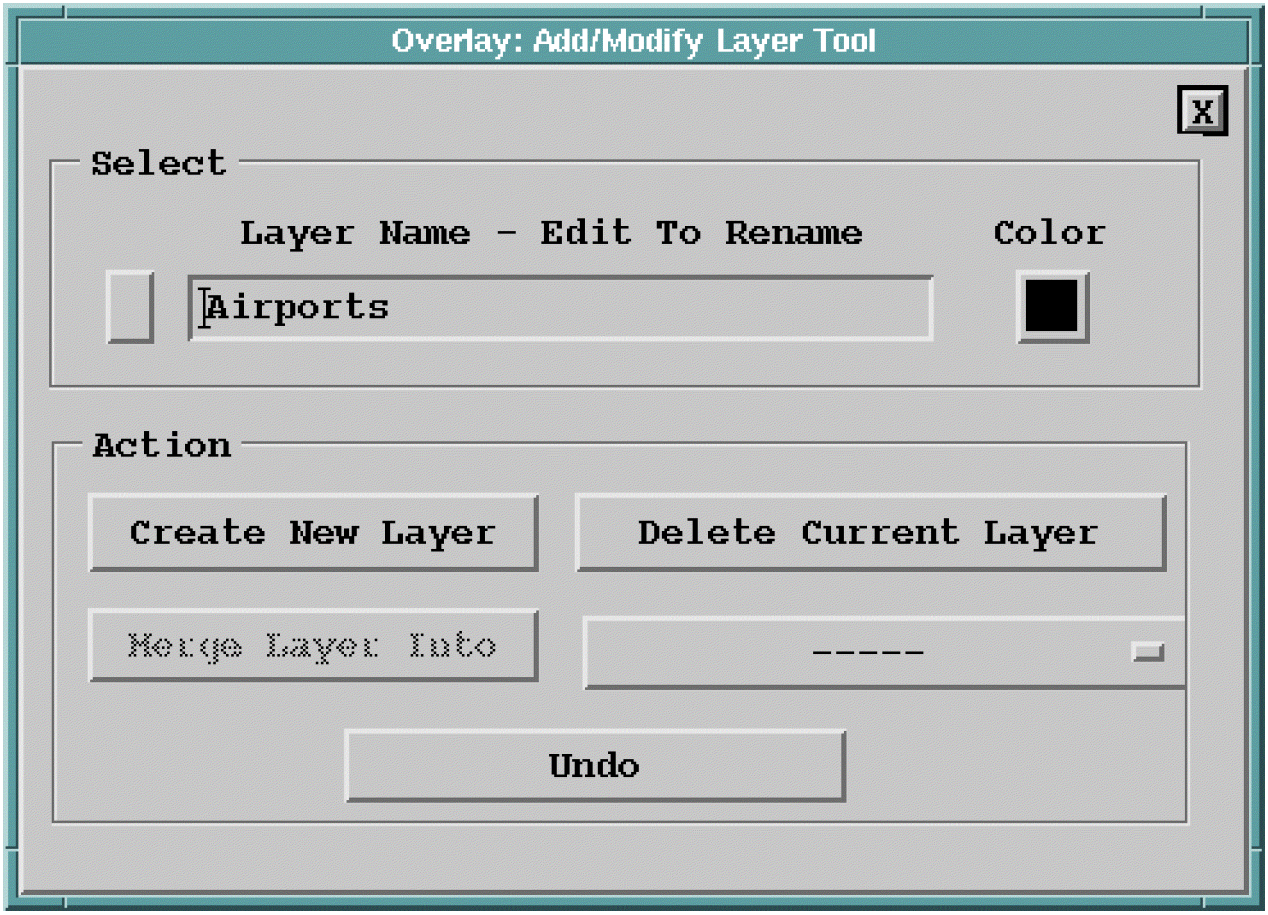
- NoUnderlay
- Protected are the areas you use with he WARN product
- The Projection menu lets you select the map projection from a menu


Projections are discussed in detail elsewhere in the manual. Typically, Azimuthal is used in one–radar applications, Mercator in composites in the tropical areas and Polar Stereographic in the composites near the poles.

Next item in the Options menu is the tool to add and delete things. You can add or delete  text, icons and fill points. Besides you can select “pointer” which gives you position of you cursor.



Each thing you add goes to a layer. Each layer is in one color, shown in a box next to the layer name. You can create more layers and modify them with the Add?Modify layer tool. Here you can select a color from predefined set. If you want to define more colors use the **color** utility. You can merge two layers by selecting the target layer to the down right bar (shown as with – – – in the figure).



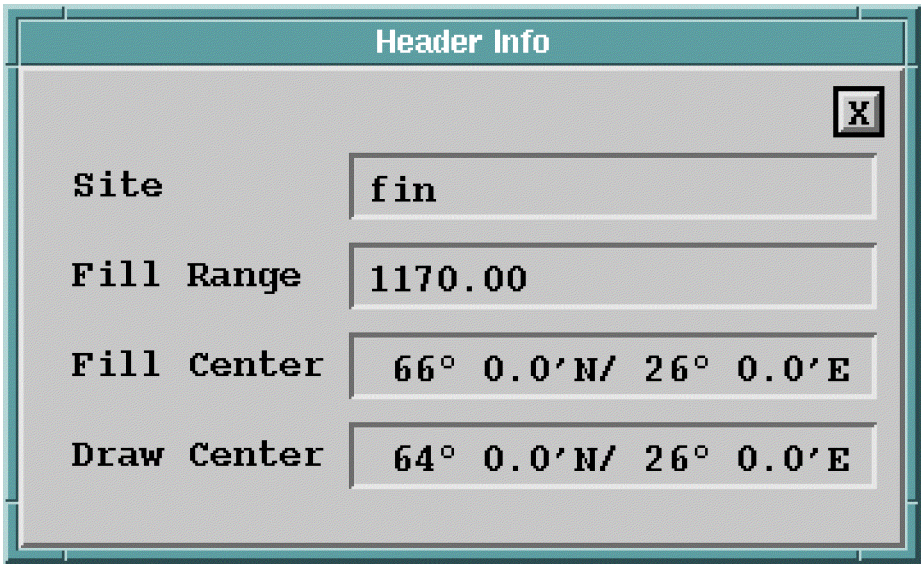

Adding icons includes a text attached to the icon, while text is just plain text. You probably want to justify text to center, so that the point shown with latitude and longitude in the upper column is under the middle character of your text. When you use icons, it's better to justify them right or left, because the icon goes to the point and the text should be next to the icon. You can highlight the text which gives it a colored background.

Fill points are used to make an underlay. An underlay is a range with outer boundaries, a filled area, and a color number specifying the color to use for the filled area. An underlay can contain up to 400 fill points. **Overlay** starts from each of the specified fill points and fills all contiguous areas, stopping at overlay lines. It is fine to enter many fill points for the same filled region. In fact, this is required for a coastline with deep bays, so that the region fills correctly at all scales.Many overlays

contain coastline and use a blue underlay for ocean. To prevent the filled color from leaking around the edge of the overlay limits, use the RANGE command to tell **overlay** how far you have drawn your overlay.

When you select a Kill option, your cursor becomes a killer tool and you can delete items from the map by pointing and clicking them. There is a separate kill tool for text, icons and fill points to minimize the risk you kill something you didn't want to kill. If you can't kill what you want, you are probably working with a wrong layer.

Last item in the options menu is a header info panel. It gives you info about the overlay file header, described at chapter 7.4.1



## 7.2 Listing and Printing Overlay Files

Overlay files in use on are stored on disk in a directory called IRIS_OVERLAY. The Product Output process always looks for overlay files in that directory.

To see a list of the overlay files stored on your system, issue the following commands from the operating system prompt:

```
$ cd $IRIS_OVERLAY
$ ls
```

Overlays for many of SIGMET's IRIS sites are included in the release in the IRIS_ROOT/config_template/overlay directory. **overlay** will default to the IRIS_OVERLAY directory, unless you specify a path in the filename.

The overlay directory should contain only overlay files. Do not use it to store any other kinds of file. In addition, SIGMET recommends that you follow these file naming conventions for any overlay files that you create:

- Use the .ovr suffix or file extension to denote overlays. The Product Output process uses only these files. The **overlay** utility uses files with this extension by default. That is, if the file has the .ovr suffix, you do not need to type the suffix when you display it using **overlay**.
- Use .xbm to denote bitmap overlay icons that are referenced within the overlay files.

Because overlays are ASCII files, you can print them on your system printer.

## 7.3 Viewing an Overlay with overlay

The overlay template directory contains a file called sample.ovr. It contains some text strings, a simple overlay area, and an underlay area. Later in this chapter, you will see how this sample overlay is defined. Look at it now using the **overlay** utility.

### To display sample.ovr:

1. Run the **overlay** utility, specifying the name of the overlay file, as follows:

```
$ cd ${IRIS_ROOT}/config_template/overlay
$ overlay ./sample
```

**overlay** processes the file (which may take several seconds), and responds as follows:

```
12 (max 40) icon files, using 311 (max 12000) bytes:
 0: dot9.xbm             1: dot7.xbm             2: dot13.xbm
 3: airport.xbm          4: beacon.xbm           5: test.xbm
 6: building.xbm         7: golf.xbm             8: hospital.xbm
 9: park.xbm            10: poi.xbm             11: train.xbm
Processing Overlay file.
10 (max 200000) points, 25 (max 400) strings, and 9 (max 4000)
fill points
1 (max 32) named layers:
 0: Default
Site name:'sample'
Draw center: 42¯24.0'N  71¯ 6.0'W
Fill center: 42¯24.0'N  71¯ 6.0'W

     Center:  42¯24.0'N,  71¯ 6.0'W
Upper right:  44¯46.0'N,  67¯18.0'W
 Lower left:  39¯55.0'N,  67¯18.0'W
```

This summarizes the icon files found, then the number of end points, text strings, and fill points in the selected overlay. Finally, the site name and the latitude and longitude of the draw and fill center are shown. **overlay** always displays centered at the draw center, while the Product Output menu aligns it with the radar.

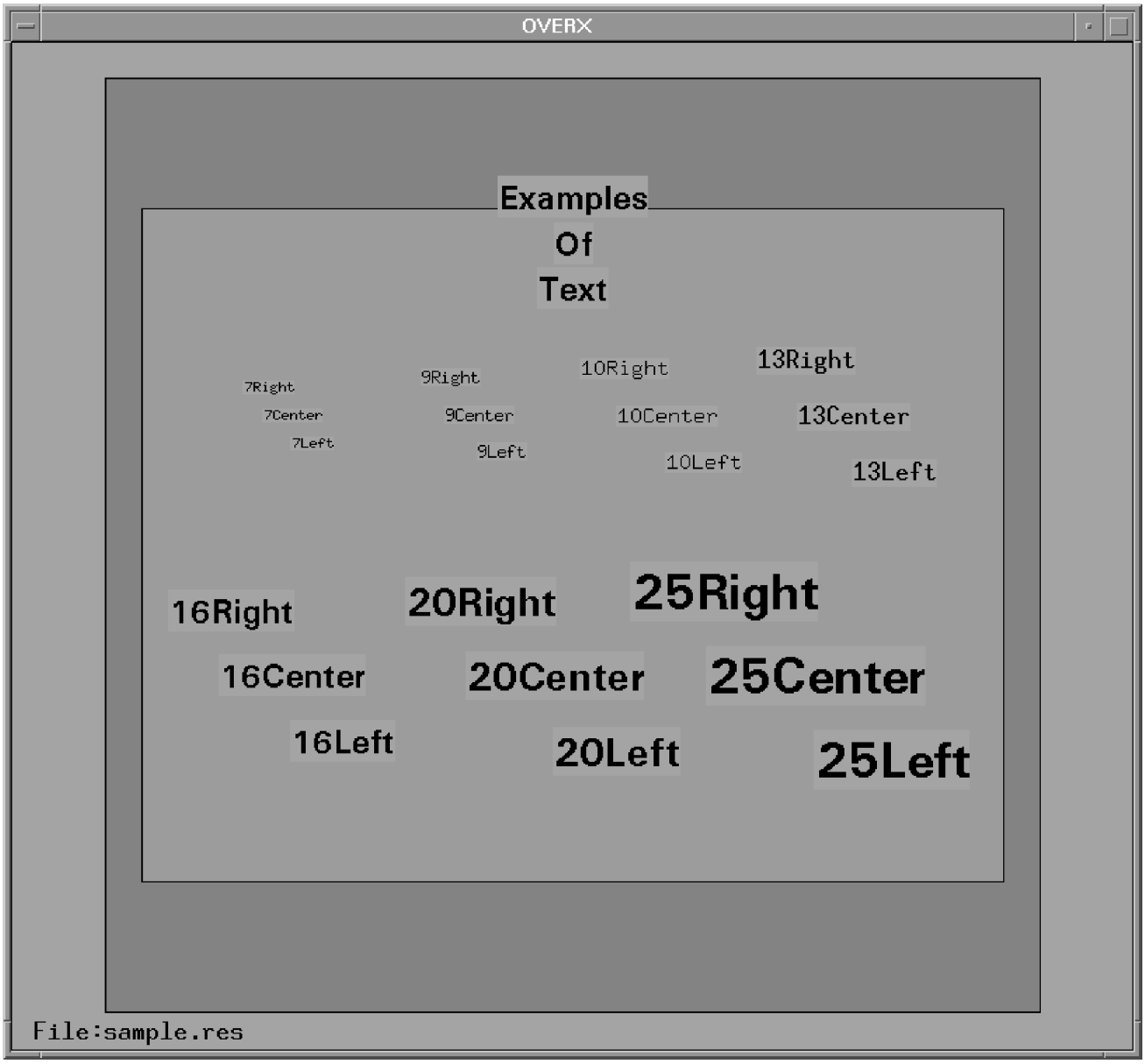
If you are editing the overlay, you may want to use a small scale and offset the radar from the center of the screen so that you can view a specific region in greater detail. Experiment with different radar locations and scales until you get exactly what you want on the display.

2. **overlay** displays the overlay on the screen in using the colors specified in **color_setup**, as shown in Figure 7–1. Optional range rings or a latitude and longitude grid are drawn in green.

   Underlays have a maximum range, which is specified in the file. A circle is drawn at that range, the underlay is filled, then the circle is removed.

3. When you have finished looking at the overlay file, return to the terminal or window where you invoked **overlay** and press CTRL/C.

**Figure 7–1: Sample Overlay Display**



# 7.4 Format of Overlay Files

When creating an overlay, you must transfer the points from a map to an overlay file. There are two major paths available to you when building an overlay file. These are either **Absolute**, or **Relative** coordinates.

Overlay files are defined with a special syntax. A small set of simple statements specify the location of the radar and place the text or lines on the map. A semicolon (;) marks the beginning of a comment line. The END statement marks the end of the map. Anything that appears in the file beyond the END statement is ignored. It is sometimes helpful during editing to put an END in the middle of a file, so that **overlay** plots only the map lines and text up to that point.

## 7.4.1 Overlay Header

If the overlay is defined in absolute coordinates all points in the overlays are given in terms of a LONGITUDE, LATITUDE coordinate pair. This of course sets an absolute position for each point. When using absolute coordinates, the following header commands must be included at the top of the overlay file:

| | |
|---|---|
| **PROJECTION latlon** | Informs system overlay is in absolute coordinates. |
| **DRAW_CENTER** ***lon lat*** | Tells **overlay** where to center the image. Generally this is the either the coordinates of the radar site, or a nearby special area of interest (i.e. an airport being monitored). |
| **FILL_CENTER** ***lon lat*** | Specifies center of fill area. Typically this is the center of the array of data points you have generated. |
| **FILL_RANGE** ***range*** | Sets the maximum range that the underlay covers, in km. **overlay** draws a circle at this distance from the fill center before filling the underlay. |

If the overlay is defined in relative coordinates, all points are defined in terms of "map units". Map units are any arbitrary unit of measure that is E/W and N/S of an origin (0,0), which can be anywhere on the map. Often millimeters from graph paper are used to designate the points on a map. The map must be in azimuthal equidistant projection, with the specified reference point. A scale factor specifies the conversion between these map units and distance on the surface of the earth. The **"RADAR x y"** command is used to specify the location of the reference point in the map units. The "**SCALE n**" command must also be used to specify how many kilometers each map unit specifies. For example, if a **RADAR 100 200** and a **SCALE 0.5** command are used, then if later a point is specified as **POINT 110 200**, this point will be exactly 5km due east of the reference position. In summary, the following headers commands must be defined at the beginning of an overlay file when using relative coordinates:

| | |
|---|---|
| **PROJECTION aed** | Informs system overlay is in relative coordinates. |
| **SCALE** ***scale*** | Sets the scale factor in km/map unit. |

| | |
|---|---|
| **RADAR** ***x y*** | Sets the projection point in map units. |
| **REFERENCE** ***lon lat*** | Sets the projection point in absolute coordinates. |
| **DRAW_CENTER** ***lon lat*** | Same as latlon projection above. |
| **FILL_CENTER** ***lon lat*** | Same as latlon projection above. |
| **FILL_RANGE** ***range*** | Same as latlon projection above. |

For backwards compatibility, the old **MIDDLE** command still has the meaning of setting both the draw center and the fill center.

## 7.4.2 Text Strings and Bitmap Icons

You can place many strings of text on the overlay. Use the full ROMAN8 character set. If you wish to include imbedded spaces, please enclose the string in double quotes (”).

You can place text on the overlay in terms of map units, specifying the justification — left, center, right, top, or bottom. The justification remains in effect for all subsequent strings until another justification is specified.

You can also choose to place text on the overlay relative to the edge of the display screen. In these cases, the position is in 10000ths of the screen measured from the lower left corner. For example, a location of (5000, 500) is in the middle near the bottom.

### Appearance statements:

| | |
|---|---|
| **SIZE #** | Set the text size in pixels, range 7 – 26. |
| **HIGHLIGHT** | Use highlighted text (reverse video). |
| **NORMAL** | Use normal video text. |

### Map units justification statements:

| | |
|---|---|
| **TOP** | Text is top-justified. |
| **BOTTOM** | Text is bottom-justified. |
| **LEFT** | Text is left-justified. |
| **RIGHT** | Text is right-justified. |
| **CENTER** | Text is centered. |

### Screen-relative justification statements:

| | |
|---|---|
| **SCN_CENTER** | Text is centered relative to the screen. |
| **SCN_LEFT** | Text left-justified relative to the screen. |
| **SCN_RIGHT** | Text is right-justified relative to the screen. |

**String location and content statements:**

| | |
|---|---|
| **TEXT** ***x y str*** | Places the text *str* at the specified location. |
| **ICON_TEXT** ***str*** | Same as above TEXT command, but *str* is placed next to the bitmap icon specified in the most recent ICON command. |
| **TEXT** ***x y str1 str2*** | Places the text *str1* at the specified locations. The text *str2* is available in the tabulation key in the display options menu. Generally *str1* is a short abbreviation that fits cleanly on the screen and *str2* is the full string that is not printed on the screen, but is available in the tabulation. |
| **ICON_TEXT** ***s1 s2*** | Same as the above TEXT command, but *s1* is placed next to the bitmap icon specified in the most recent ICON command. |

**Icon command:**

| | |
|---|---|
| **ICON** ***x y name*** | Places bitmap icon specified by *name* at position *x y.* |

The bitmap icons that iris supports are black and white icons (.xbm files). Color icons are not supported (.xpm files). Icons can be edited using the icon editor that comes with the UNIX system. Icons of any size are supported by iris, but using large icons would not be practical. The user will find that as icons get larger than 16 by 16 pixels, they tend to take up too much screen real estate. Once icons are edited in the icon editor, the xbm files should be placed in the $IRIS_OVERLAY directory and overlay files can make reference to them by name.

## 7.4.3 Map Outlines

An overlay typically consists of a lot of lines. You draw a line by specifying the beginning and ending points. The GAP statement denotes a break in the line.

**Outline statements:**

| | |
|---|---|
| **POINT** ***x y or x y*** | Places a point at the specified coordinates along an overlay line; expressed in map units. |
| **GAP** ***or*** ***** | Marks the end of an overlay line. |

## 7.4.4 Layer Functions and Command

Overlay files can be composed of a single layer, or divided up into up to 32 different layers. When only a single layer is used, the complete overlay is always displayed (this includes all lines, text and icons). When multiple layers are defined, any combination of any number of the layers can be displayed at runtime. For example, there may be a main layer called coastlines that is always displayed, then there may

be other layers such as roads, rivers, power plants, etc. that can be turn on or off individually. Each layer can contain lines, text and icons, but does not need to contain all three. When defining layers in an overlay file, all entries in the file that occur after the initial layer command are applied to that layer. This continues sequentially through the file until either the file ends, or another layer command is encountered. Below is the summary of the two command associated with layer definitions:

**LAYER *name*** This defines a layer called *name.* All commands following in the overlay file (until another LAYER command is encountered) are applied this this layer. This layer can be enabled at run time by selecting its *name* in the display options menu.

**LAYER_COLOR *name color*** All lines defined in the layer called *name* will be drawn in the color *color.* Color should be either overlay1, overlay2, or overlay3. These colors are defined in **color_setup**.

## 7.4.5 Solid Underlay Regions

An underlay is a range with outer boundaries, a filled area, and a color number specifying the color to use for the filled area. We use the term “underlay” to indicate that this color is visible only when there is thresholded radar data, so it seems to be under the data. **Overlay** starts from each of the specified fill points and fills all contiguous areas, stopping at overlay lines. It is fine to enter many fill points for the same filled region. In fact, this is required for a coastline with deep bays, so that the region fills correctly at all scales.

Many overlays contain coastline and use a blue underlay for ocean. To prevent the filled color from leaking around the edge of the overlay limits, use the RANGE command to tell **overlay** how far you have drawn your overlay.

### Underlay statements:

**FILL_COLOR *color*** Sets the color of the underlay region. Color should be either underlay2 or underlay3.

**FILL *x y*** Marks the starting point for the fill area.

## 7.4.6 GIF Underlay Regions

A GIF underlay is a full color picture used to fill the background between overlay lines and features. Like solid underlays, this is visible only when there is thresholded radar data. Areas outside the radar coverage circle are displayed with a darkened version of the GIF underlay.

Be careful not to mix GIF underlays and filled solid underlays.

Available projection choices include: AED, MERCATOR, POLSTERIO, UTM, or PERSPECTIVE.

### GIF Underlay statements:

| | |
|---|---|
| **UNDERLAY_CENTER *lon lat*** | Selects the center location of an underlay file. |
| **UNDERLAY_PROJECTION *AED*** | Selects the projection of the underlay file. |
| **UNDERLAY_REFERENCE *lon lat*** | Selects the projection reference location. |
| **UNDERLAY_SCALE x *y*** | Selects the projection scale in km/pixel. |
| **UNDERLAY_FILE *file.gif*** | Selects the name of an underlay file. The file must resi in the ${IRIS_OVERLAY} directory. |

### To build a gif underlay:

You can download a GIF underlay from the web. To do this, go to http://geoengine.nima.mil. Click the "Define AOI" (Area Of Interest) tab. Then define the corner latitudes and longitudes. Start for example with a 10 degree by 10 degree area. Then click the "Download/Order" tab. Select DTED LEVEL 0, JPEG, and Windows. This will download to you a zipped jpeg image. You can view this directly using xv, and you can incorporate it into your overlay as described below.

The first thing you need to do it convert the downloaded JPEG file to GIF format. You can do this with the linux xpaint program. If is is not installed on your system, it is available as a separate rpm file. Be sure to get the filename extension at ".gif". While in xpaint, you will also want to convert the water to blue. You will find that the ocean has some speckles near the edge, which you can manually edit. Coastline details and flat low countries will require manual repair. Areas far out to sea, away from land, will arrive as white, and need to be repaired. While editing, check the gif scale. It is usually 120 pixels/degree. Compute this by taking the width of the gif image in pixels minus 1 and dividing by the width in degrees.

For images near the poles (such as Sweden), the image may arrive compressed horizontally by a factor of 2, so you will need to repair that. There is also a maximum pixel size supported, and larger areas tend to degrade in resolution, so you may need to download several images and attach them together.

The image is generated in a gray scale. SIGMET recommends that you change to your favorite background color, we suggest a sandy brownish. This will show up better with satellite images, for example. You can use the PC lview program's "color balance" operation for this.

To install the gif image in your overlay:

1) Copy your gif file to the overlay directory.

2) Create a new layer called "Underlay" using the add/delete menu. Go back to the "Options" menu to turn on the new Underlay layer, so you will see it later.

3) Select action "add underlay" in the add/delete menu. Pick the Underlay layer, then fill in everything in the underlay properties box. The projection type is "Equidist Cylinder". The center and projection reference should be the center of your image.

Sometimes the downloaded image seems to be 1 degree or 1 minute off from the expected value. To compute the underlay scale, first compute the range scale, which is the the number of km/degree at your center lattitude. Use the following equation:

$$RangeScale = \frac{\pi}{180} \times 6371km \times \cos(\ Latitude\ )$$

Finally divide by the gif scale (in pixels/degree) to give you the underlay scale in km/pixel. For example, an overlay centered at 45 degrees north, will have a range scale of 78.62 km/degree, and at 120 pixels/degree will yield 0.65522 km/pixel.

4) Apply and save.

## 7.4.7 Example of an Overlay File

```
;  This is a sample overlay to demonstrate the overlay features.
;  Put this in ${IRIS_ROOT}config/overlay and call it sample.ovr
;
;  The semicolon ";" is used at the beginning of comment lines.
;
;  All distances are in arbitrary "map units" with coordinate points
;  denoted with E-W first and N-S second.  See overlay description.
;
PROJECTION AED
;  The scale factor in km per "map unit":
SCALE   1.000
;
;  The radar location in map units:
RADAR 0 0
REFERENCE -71.100 42.400
;
SITE  sample
FILL_CENTER  -71.100 42.400
DRAW_CENTER  -71.100 42.400

;  Next come the text strings:
;      First the text size (Height in pixels, range 7 through 25)
SIZE 16
;
;      Next the text justification (left, center or right)
SCN_CENTER
;
;      Then the text location and text string (E-W first, N-S second)
HIGHLIGHT
TEXT 5000 8450  Examples
TEXT 5000 8000  Of
TEXT 5000 7550  Text
;
;  Here are font and justification examples for many sizes
;  First do the size 7
SIZE 7
RIGHT
TEXT -150  85 7Right
;  The two lines above cause the text "7Right" to be positioned with
; its right at map unit position (250, 350).
CENTER
TEXT -150  70 7Center
;  For center, there is no position dot and text "7Center" is
centered
;  at the specified location.
LEFT
TEXT -150  55 7Left
;  For left, the dot is to the left of the text "7Left"
;
SIZE 9
RIGHT
TEXT  -50  90 9Right
CENTER
TEXT  -50  70 9Center
LEFT
TEXT  -50  50 9Left
```

```
;
;  Now change the size to 10 and repeat the examples:
SIZE 10
RIGHT
TEXT   50  95 10Right
CENTER
TEXT   50  70 10Center
LEFT
TEXT   50  45 10Left
;  Now change the size to 13 and repeat the examples:
SIZE 13
RIGHT
TEXT  150 100 13Right
CENTER
TEXT  150  70 13Center
LEFT
TEXT  150  40 13Left
;  Now change the size to 16:
SIZE 16
RIGHT
TEXT –150  –35 16Right
CENTER
TEXT –150  –70 16Center
LEFT
TEXT –150 –105 16Left
SIZE 20
RIGHT
TEXT   –10  –30 20Right
CENTER
TEXT   –10  –70 20Center
LEFT
TEXT   –10 –110 20Left
SIZE 25
RIGHT
TEXT  130  –25 25Right
CENTER
TEXT  130  –70 25Center
LEFT
TEXT  130 –115 25Left
;
;  Here is a message in the lower left corner of the screen
SIZE 13
SCN_LEFT
TEXT 200 200 File:sample.ovr

;  And finally the points for the map lines in map units:
;  Example of a small rectangle to enclose text examples:
;  E–W first, N–S second.
–230 –180
–230  180
 230  180
 230 –180
–230 –180
GAP
;  The word ”gap” above is used to denote the last point of a map
line.
;
;  To start another map line, continue entering map coordinates and
```

```
;  put ”GAP” after the line is complete.
;
; Put ”END” at the end of the file.  Note that it is sometimes
; helpful during editing to put ”END” in the middle of a file
; so that overlay will only plot the map lines and text up to
; that point.
;
;  Here is a large rectangle around everything:
–250 –250
–250  250
 250  250
 250 –250
–250 –250

;  Here are some underlay fill points.  These points are used to
start a
;  region fill, and should be spaced about the area of concern.
RANGE 0
FILL_COLOR underlay2
FILL –240 –200
FILL –240 –100
FILL –240   50
FILL –240  150
;
FILL 240 –150
FILL 240  –50
FILL 240   50
FILL 240  150
FILL_COLOR underlay3
FILL –50 –50
END
;  End of SAMPLE
```

## 7.5 Format of catchment files

Catchment files are similar in format to overlay files. They contain a series of points connected by vectors. They also support a ";' for comments, and "END" to mark the end of the file. There is only one file header command:

**DRAW_CENTER *lon lat*** Tells **overlay** where to center the image. Generally this is the either the coordinates of the radar site, or the center of the catchment area.

**SIZE *size*** Sets the size to draw subcatchment names. This is the height in pixels, and it applies until the next size command in the file.

Catchment files are divided into a series of subcatchments. Each one starts with a header specifying the name and label location. Following this header, are the data points. They must define a closed polygon. Make sure the last point matches the first one. There can be a maximum of 512 subcatchments defined in each catchment file.

**START *number name*** Tells the number and name of the subcatchment area

**LABEL *lon lat*** Specifies the location to draw the name when labeling the subcatchment. This location must be inside the region.

IRIS supports a maximum of 20 catchment files.

# 7.6 Creating and Editing Overlay Files

**NOTE: Creating and editing overlay files should be done only under the supervision of the system manager.**

### To create an overlay file:

1. Use a computerized geographic database to get the array of coastlines and political boundaries you want.
2. Manually add the header commands, display it with overlay and make needed repairs.
3. Add the desired text and icon features.
4. Add underlay fill points. When doing this you may discover more breaks in the overlay lines which need to be repaired.
5. Test the underlay filling by displaying over a broad range of image sizes, for example 50, 100, 200, 400, and 800 km. It is usually necessary to add fill points in lakes and bays pointing towards the display center.
6. When you are satisfied with your overlay, place it in the $IRIS_OVERLAY directory.