Chapter 6

Vector Maps

This chapter provides an explanation of how vector map layers are accommodated in the GRASS database.

6.1. What is a Vector Map Layer?

GRASS vector maps are stored in an *arc-node* representation, consisting of nonintersecting curves called *arcs.* An arc is stored as a series of x,y coordinate pairs.[1] The two endpoints of an arc are called *nodes.* Two consecutive x,y pairs define an arc *segment.* [2]

The arcs, either singly, or in combination with others, form higher level map features: *lines* [3] (e.g., roads or streams) or *areas* [4] (e.g., farms or forest stands). Arcs that form linear features are sometimes called *lines,* and arcs that outline areas are called *area edges* or *area lines.* [5]

Each map feature is assigned a single integer attribute value called the *category* number. For example, assume a vector file contains land cover information for a state park. One area may be assigned category 2 (perhaps representing prairie); another is assigned category 3 (for forest); and so on. Another vector file which contains road information may have some roads assigned category 1 (for paved roads); other roads may be assigned category 2 (for gravel roads); etc. See *§5.1 What is a Raster Map Layer?* [*p. 23*] for more information about GRASS category values.

A vector map layer is stored in a number of data files. The files which comprise a single

1 For this reason *arcs* are also called *vectors.*

2 Arc *segments* are sometimes called *line-segments.*

3 A *line* here does not mean a straight line between two points. It only means a linear feature.

4 *Areas* are also called *polygons.* The GRASS vector format does not store the polygons explicitly. They are constructed by finding the particular *arcs* which form the polygon perimeter.

5 Obviously, there is some confusion in the GIS vector terminology. This is partly due to use of terms that have a common meaning as well as a mathematical meaning. Vector terminology is a subject for much debate in the GIS world.

paste vect.xfig diagram here

vector map layer all have the same name, but each resides in a different database directory under the mapset.[6] These database directories are:

directory	function
dig	binary arc file
dig_ascii	ascii arc file
dig_att	vector category attribute file
dig_cats	vector category labels
dig_plus	vector index/pointer file
reg	digitizer registration points

For example, a map layer named *soils* would have the files *dig/soils, dig_att/soils, dig_plus/soils, dig_ascii/soils, dig_cats/soils, reg/soils,* etc.

Note. Vector files are also called *digit* files, since they are created and modified by the GRASS digitizing program *v.digit.*

Note. When referring to one of the vector map layer files, the directory name is used. For example, the file under the *dig* directory is called the *dig* file.

Note. Library routines which read and write vector files are described in *§13 Vector Library* [*p. 157*].

6.2. Ascii Arc File Format

The arc information is stored in a binary format in the *dig* file. The format of this file is reflected in the ascii representation stored in the *dig_ascii* file. It is the ascii version which is described here.[7]

---

[6] Database directories are also called *elements.* See *§4.4 Mapsets* [*p. 16*] for a description of database elements.

[7] The programs *v.import, v.in.ascii,* and *v.out.ascii* convert between the ascii and binary formats.

The *dig_ascii* file has two sections: a header section, and a section containing the arcs.

6.2.1. Header Section

The header contains historical information, a description of the map, and its location in the universe. It consists of fourteen entries. Each entry has a label identifying the type of information, followed by the information. The format of the header is:

label	format	description
ORGANIZATION:	text (max 29 characters)*	organization that digitized the data
DIGIT DATE:	text (max 19 characters)*	date the data was digitized
DIGIT NAME:	text (max 19 characters)*	person who digitized the data
MAP NAME:	text (max 40 characters)*	title of the original source map
MAP DATE:	text (max 10 characters)*	date of the original source map
OTHER INFO:	text (max 72 characters)*	other comments about the map
MAP SCALE:	integer	scale of the original source map
ZONE:	integer	zone of the map (e.g., UTM zone)
WEST EDGE:	real number (double)	western edge of the entire map †
EAST EDGE:	real number (double)	eastern edge of the entire map †
SOUTH EDGE:	real number (double)	southern edge of the entire map †
NORTH EDGE:	real number (double)	northern edge of the entire map †
MAP THRESH:	real number (double)	digitizing resolution ‡
VERTI:	(no data)	marks the end of the header section

The labels start in column 1 and continue through column 14. Labels are uppercase, left justified, end with a colon, and blank padded to column 14. The information starts in column 15. For example:

```
ORGANIZATION: US Army CERL
DIGIT DATE:   03/18/88
DIGIT NAME:   grass
MAP NAME:     Urbana, IL.
MAP DATE:     1975
OTHER INFO:   USGS sw/4 urbana 15' quad. N4000-W8807.5/7.5
MAP SCALE:    24000
ZONE:         16
WEST EDGE:    383000.00
EAST EDGE:    404000.00
SOUTH EDGE:   4429000.00
NORTH EDGE:   4456000.00
MAP THRESH:   0.00
VERTI:
```

* Currently, GRASS programs which read the header information are not tolerant of text fields which exceed these limits. If the limits are exceeded, the ascii to binary conversion will probably fail.

† The edges of the map describe a region which should encompass all the data in the vector file.

‡ The MAP THRESH is set by the *v.digit* program. If the data comes from outside GRASS, this field can be set to 0.0.

 - 36 -

6.2.2. Arc Section

The arc information appears in the second section of the *dig_ascii* file (following *VERTI:* which marks the end of the header section). Each arc consists of a description entry, followed by a series of coordinate pairs. The description specifies both the type of arc (*A* for area edge, or *L* for line[8]), and the number of points (coordinate pairs) in the arc. Then the points follow.

For example:

```
A  5
  4434456.04   388142.16
  4434446.65   388202.64
  4434407.49   390524.38
  4434107.06   390523.59
  4433326.51   390526.48
L  3
  4434862.31   392043.33
  4434872.42   394662.14
  4434871.44   398094.75
A  3
  4454747.38   396579.60
  4454722.69   393539.73
  4454703.68   390786.90
```

In this example, the first arc is an area edge and has 5 points. The second arc is part of a linear feature and has 3 points. The third arc is another area edge and has 3 points.

The arc description has the letter *A* or *L* in the first column, followed by at least one space, and followed by the number of points.[9]

Point entries start with a space, and have at least one space between the two coordinate values.[10]

Note. The points are stored as **y,x** (i.e., north, east), which is the reverse of the way GRASS usually represents geographic coordinates.

Note. If the *v.digit* program has deleted an arc, the arc type will be represented using a lower case letter (i.e., *l* instead of *L, a* instead of *A*). Of course, this will only be manifest when a binary *dig* file with a deleted arc is converted to the ascii *dig_ascii* file.

---

[8] Other types may be added in the future.

[9] This can be written with the Fortran format : *A1,1X,I4.*

[10] These can be written with the Fortran format : *2(1X,F12.2).*

6.3. Vector Category Attribute File

As was mentioned in *§6.1 What is a Vector Map Layer?* [p. 33], each feature in the vector map layer has a *category* number assigned to it. The category number for each map feature is not stored in the *dig* file itself, but in the *dig_att* file.

The *dig_att* file is an ascii file that has multiple entries, each with the same format. Each entry refers to one map feature, and specifies the feature type (area or line), an x,y marker, and a category number.

For example:

```
A    389668.32    4433900.99    7
L    395103.96    4434881.19    2
```

In this example, an area feature is assigned category 7, and a linear feature is assigned category 2.

The x,y marker is used to find the map feature in the *dig* file. It must be located so that it uniquely identifies its related map feature. In particular, an area marker must be inside the area, and a line marker must be closer to its related line than to any other line (preferably on the line) and not at a node.

If multiple entries identify the same map feature, only one will be used (currently thee last entry).

A map feature which has no entry in this file is considered to be unlabeled. This means that during the vector to raster conversion (i.e., *v.to.rast*), unlabeled areas will convert as category zero, and unlabeled lines will be ignored.

The format of this file is rather strict, and is described in the following table:

columns	data
1	Type of map feature (*A* or *L*)*
2-3	spaces
4-15	Easting (x) of the marker, right justified
16-17	spaces
18-29	Northing (y) of the marker, right justified
30-31	spaces
32-39	Category number, right justified
40-49	spaces
50	newline †

* Other types, such as *point,* may be allowed in the future.

† UNIX text files are terminated with a newline. Therefore, each entry will appear as 49 characters. The entire file size should be a multiple of 50.

This format is required by programs which modify the vector map (e.g., *v.digit*). Programs which only read the vector map accept a looser format: the feature type must start in column 1; the items must be separated by at least one space; and the entries must be less than 50 characters. Also, the program *v.support* will convert the looser format to this stricter format.

Note. The marker is specified as **x,y** (i.e., east, north), which is the way GRASS usually represents geographic coordinates, but which is reverse of the way the arcs are stored in the *dig_ascii* file.

6.4. Vector Category Label File

Each category in the vector map layer may have a one-line description. These category labels are stored in the *dig_cats* file. The format of this file is identical to the raster category file described in *§5.4 Raster Category File Format* [p. 28], and the reader is referred to that section for details.

Note. The program *v.support* allows the user to enter and modify the vector category labels. The program *v.to.rast* copies the *dig_cats* file to the raster category file during the vector to raster conversion.

Note. Library routines which read and write the *dig_cats* file are described under *§12.11.6 Vector Category File* [p. 125].

6.5. Vector Index and Pointer File

The *dig_plus* file contains information that accelerates vector queries. It is created by the program *build.vect* (which is run by *v.digit* when a vector file is created or modified, and by *v.support* at user request) from the data in the *dig* and *dig_att* files.

For this reason, and since the internal structure of the *dig_plus* file is complex, the format of this file will not be described.

6.6. Digitizer Registration Points File

The *reg* file is an ascii file used by the *v.digit* program to store map registration control points. Each map registration point has one entry with the easting and northing of the map control point. For example:

 - 39 -

383000.000000	4429000.000000
383000.000000	4456000.000000
404000.000000	4456000.000000
404000.000000	4429000.000000

Note. This file is used by *v.digit* only. It is *not* used by any other program in GRASS.

6.7. Vector Topology Rules

The following rules apply to the vector data:

1 Arcs should not cross each other (i.e., arcs which would cross must be split at their intersection to form distinct arcs).

2 Arcs which share nodes must end at exactly the same points (i.e., must be *snapped* together). This is particularly important since nodes are not explicitly represented in the arc file, but only implicitly as endpoints of arcs.

3 Common boundaries should appear only once (i.e., should not be double digitized).

4 Areas must be explicitly closed. This means that it must be possible to complete each area by following one or more area edges that are connected by common nodes, and that such tracings result in closed areas.

5 It is recommended that area features and linear features be placed in separate layers. However if area features and linear features must appear in one layer, common boundaries should be digitized only once. An area edge that is also a line (e.g., a road which is also a field boundary), should be digitized as an area edge (i.e., arc type *A*) to complete the area. The area feature should be labeled as an area (i.e., feature type *A* in the *dig_att* file). Additionally, the common boundary arc itself (i.e., the area edge which is also a line) should be labeled as a line (i.e., feature type *L* in the *dig_att* file) to identify it as a linear feature.

6.8. Importing Vector Files Into GRASS

The following files are required or recommended for importing vector files from other systems into GRASS:

dig_ascii

The *dig_ascii* file, described in *§6.2 Ascii Arc File Format* [p. 34], is required.

dig_att

The *dig_att* file, described in *§6.3 Vector Category Attribute File* [p. 37], is essentially required. While the *dig_ascii* file alone is sufficient for simple vector display, the *dig_att* file is required for vector to raster conversion, as well as more

 - 40 -

sophisticated vector query.

dig_cats

The *dig_cats* file, described in *§6.4 Vector Category Label File* [*p. 38*], while not required, allows map feature descriptions to be imported as well.

Note. The *dig_plus* file, described in *§6.5 Vector Index and Pointer File* [*p. 38*], is created by the GRASS program *v.import* when converting the *dig_ascii* file to the binary *dig* file.