Next: VECT, Previous: BBP and BEZ, Up: Object File Formats

4.2.5 OFF Files

The conventional suffix for `OFF` files is `.off`.

Syntax:

```
[ST][C][N][4][n]OFF     # Header keyword
[Ndim]        # Space dimension of vertices, present only if nOFF
NVertices  NFaces  NEdges   # NEdges not used or checked

x[0]  y[0]  z[0]     # Vertices, possibly with normals,
          # colors, and/or texture coordinates, in that order,
          # if the prefixes N, C, ST
          # are present.
          # If 4OFF, each vertex has 4 components,
          # including a final homogeneous component.
          # If nOFF, each vertex has Ndim components.
          # If 4nOFF, each vertex has Ndim+1 components.
...
x[NVertices-1]  y[NVertices-1]  z[NVertices-1]

                  # Faces
                  # Nv = # vertices on this face
                  # v[0] ... v[Nv-1]: vertex indices
                  #        in range 0..NVertices-1
Nv  v[0] v[1] ... v[Nv-1]  colorspec
...
                  # colorspec continues past v[Nv-1]
                  # to end-of-line; may be 0 to 4 numbers
                  # nothing: default
                  # integer: colormap index
                  # 3 or 4 integers: RGB[A] values 0..255
          # 3 or 4 floats: RGB[A] values 0..1
```

`OFF` files (name for "object file format") represent collections of planar polygons with possibly shared vertices, a convenient way to describe polyhedra. The polygons may be concave but there's no provision for polygons containing holes.

An `OFF` file may begin with the keyword `OFF`; it's recommended but optional, as many existing files lack this keyword.

Three ASCII integers follow: *NVertices*, *NFaces*, and *NEdges*. Thse are the number of vertices, faces, and edges, respectively. Current software does not use nor check *NEdges*; it needn't be correct but must be present.

The vertex coordinates follow: dimension * *Nvertices* floating-point values. They're implicitly numbered 0 through *NVertices*-1. dimension is either 3 (default) or 4 (specified by the key character `4` directly before `OFF` in the keyword).

Following these are the face descriptions, typically written with one line per face. Each has the form

```
N  Vert1 Vert2 ... VertN  [color]
```

Here *N* is the number of vertices on this face, and *Vert1* through *VertN* are indices into the list of vertices (in the range 0..*NVertices*-1).

The optional *color* may take several forms. Line breaks are significant here: the *color* description begins after *VertN* and ends with the end of the line (or the next # comment). A *color* may be:

nothing
: the default color

one integer
: index into "the" colormap; see below

three or four integers
: RGB and possibly alpha values in the range 0..255

three or four floating-point numbers
: RGB and possibly alpha values in the range 0..1

For the one-integer case, the colormap is currently read from the file `cmap.fmap` in Geomview's `data` directory. Some better mechanism for supplying a colormap is likely someday.

The meaning of "default color" varies. If no face of the object has a color, all inherit the environment's default material color. If some but not all faces have colors, the default is gray (R,G,B,A=.666).

A `[ST][C][N][n]OFF BINARY` format is accepted; See Binary format. It resembles the ASCII format in almost the way you'd expect, with 32-bit integers for all counters and vertex indices and 32-bit floats for vertex positions (and texture coordinates or vertex colors or normals if `COFF`/`NOFF`/`CNOFF`/`STCNOFF`/etc. format).

Exception: each face's vertex indices are followed by an integer indicating how many color components accompany it. Face color components must be floats, not integer values. Thus a colorless triangular face might be represented as

```
int int int int int
3   17   5   9   0
```

while the same face colored red might be

```
int int int int int float float float float
 3  17   5   9   4   1.0   0.0   0.0   1.0
```