

## Legend

### Node

Leaf

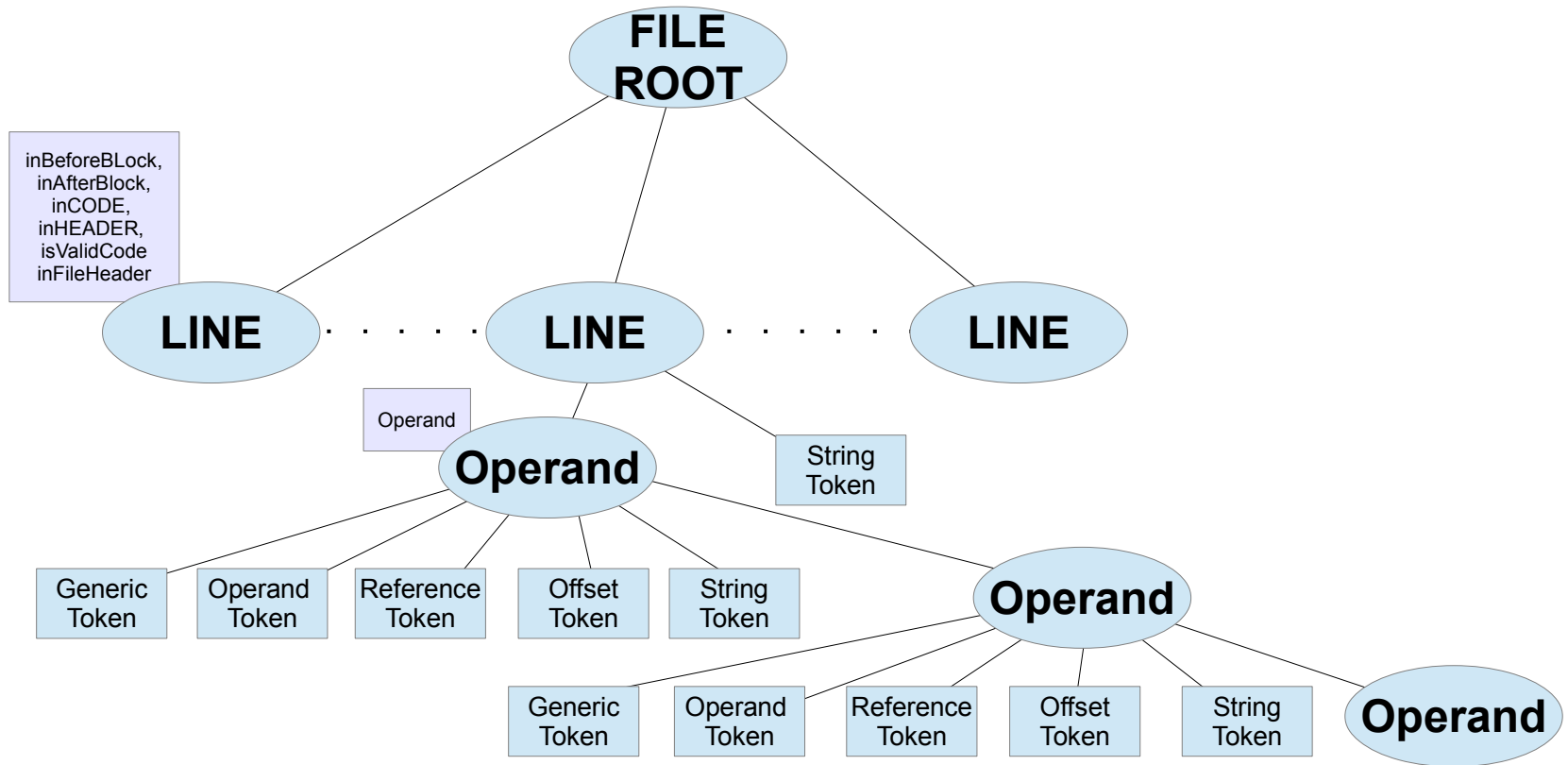
Tags

Delimiters

## MODFILE FORMAT V4

**Document**  
*extends*  
Plain/StyledDocument

MODFILEVersion  
UPKFILE  
GUID  
FUNCTION



## Legend

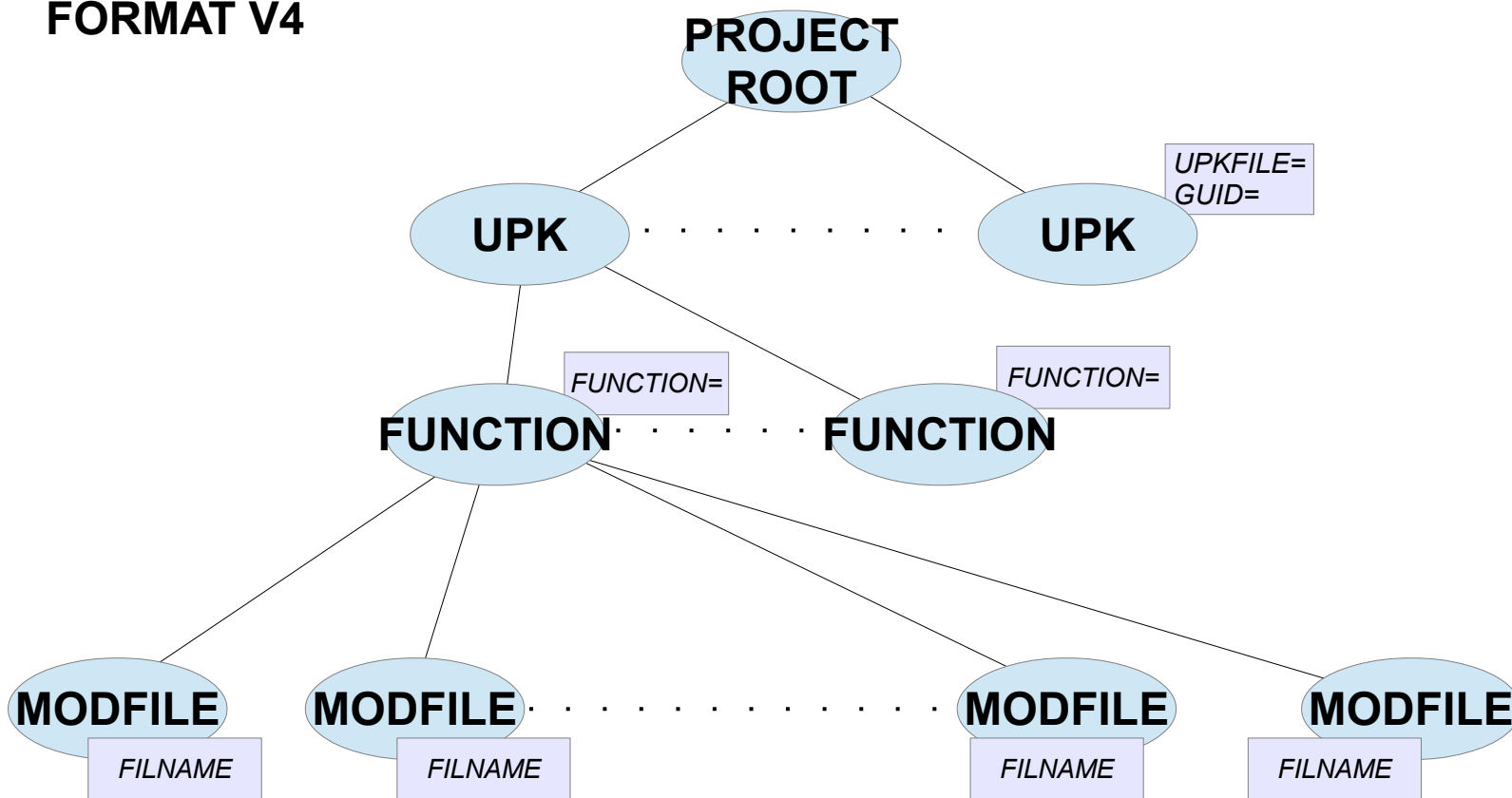
**Node**

Leaf

*Tags*

Delimiters

## MODPROJECT FORMAT V4



## Project-level categories

### PACKAGE ROOT

The root node of the package. Only one can exist. Has no owner.

### UPK

UPK Node. Begins with UPKFILE=. Lasts until next UPKFILE= . Used to determine reference mappings and upks to apply/revert to.

### FUNCTION

Function Node. Begins with FUNCTION= . Lasts until next FUNCTION= . Used to determine function scope when apply/revert hex changes.

### FILE

A single modfile I the modpackage

## File-level categories

### FILE ROOT

The root node of the file. Only one per file. Has no owner.

### LINE

Line Node. Scope from EOL to next EOL. Can represent a comment, ModFile delimiter, or line of unreal bytecode.

### OPERAND

Operand Node. Determined by parsing unreal bytecode. Represents a single byte executable command.

**Tokens are leaves containing displayable data. They may contain meta-data.**

### Operand Token

Operand token. Always 1 bytecode. Represents an operand such as 0x00, 0x01, 0x1B, or 0x19.

### Reference Token

Reference token. Always 4 bytes of bytecode. Represents a reference – an index into the objectlist, importlist, or namelist. Contains metadata indicating if it is a virtual function reference.

### Offset Token

Offset Token. Always 2 byte of bytecode. Represents an absolute or relative offset. Contains metadata indicating whether it is absolute or relative.

### String Token

String Token. Variable length. Can represent an unreal string (00 terminated) or a modfile comment (EOL terminated). Unreal strings have Operand parent nodes while modfile comments do not.

### Generic Token

Generic Token. Variable length. Can represent unreal data values (ints, floats, vectors) of fixed length or extra format-required tokens.