FEAs

FEAs are complete name/value pairs. In order to simplify and speed up scanning and processing of these names, they are represented as length-preceded data. FEAs are defined as follows:

```
struct FEA {
    unsigned char fEA;        /* byte of flags      */
    unsigned char cbName;     /* length of name     */
    unsigned short cbValue;   /* length of value    */
    unsigned char szName[];   /* ASCIIZ name        */
    unsigned char aValue[];   /* free format value  */
};
```

There is only one flag defined in fEA. That is 0x80 which is fNeedEA. Setting the flag marks this EA as needed for the proper operation on the file to which it is associated. Setting this bit has implications for access to this file by old applications, so it should not be set arbitrarily.

If a file has one or more NEED EAs, old applications are not allowed to open the file. For DOS mode applications to access files with NEED EAs, they must have the EA bit set in their exe header. For OS/2 mode, only applications with the NEWFILES bit set in the exe header may open files with NEED EAs.

Programs that change EAs should preserve the NEED bit in the EAs unless there is a good reason to change it.

The name length does not include the trailing NUL. The maximum EA name length is 255 bytes. The minimum EA name length is 1 byte. The characters that form the name are legal filename characters. Wildcard characters are not allowed. EA names are case-insensitive and should be uppercased. The FSD should call FSH_CHECKEANAME and FSH_UPPERCASE for each EA name it receives to check for invalid characters and correct length, and to uppercase it.

The FSD may not modify the flags.

A list of FEAs is a packed set of FEA structures preceded by a length of the list (including the length itself) as indicated in the following structure:

```
struct FEAList {
    unsigned long cbList;     /* length of list     */
    struct FEA list[];        /* packed set of FEAs */
};
```

FEA lists are used for adding, deleting, or changing EAs. A particular FSD may store the EAs in whatever format it desires. Certain EAs may be stored to optimize retrieval.