OS/2 and DOS Extended BootStructure and BIOS Parameter Block

The Extended Boot structure is as follows:

```
struct Extended_Boot {
    unsigned char Boot_jmp[3];
    unsigned char Boot_OEM[8];
    struct Extended_BPB Boot_BPB;
    unsigned char Boot_DriveNumber;
    unsigned char Boot_CurrentHead;
    unsigned char Boot_Sig = 41; /* Indicate Extended Boot */
    unsigned char Boot_Serial[4];
    unsigned char Boot_Vol_Label[11];
    unsigned char Boot_System_ID[8];
};
```

Where

Boot_Serial	is the 32-bit binary volume serial number for the media.
Boot_System_ID	is an 8-byte name written when the media is formatted. It is used by FSDs to identify their media but need not be the same as the name the FSD exports via FS_NAME and is NOT the name users employ to refer to the FSD. (They may, however, be the same names).
Boot_Vol_Label	is the 11-byte ASCII label of the disk/diskette volume. FAT file systems must ALWAYS use the volume label in the root directory for compatibility reasons. An FSD may use the one in the boot sector.

The extended BPB structure is a super-set of the conventional BPB structure, as follows:

```
struct Extended_BPB {
    unsigned short BytePerSector;
    unsigned char SectorPerCluster;
    unsigned short ReservedSectors;
    unsigned char NumberOfFats;
    unsigned short RootEntries;
    unsigned short TotalSectors;
    unsigned char MediaDescriptor;
    unsigned short SectorsPerFat;
    unsigned short SectorsPerTrack;
    unsigned short Heads;
    unsigned long HiddenSectors;
    unsigned long Ext_TotalSectors;
};
```

From:
<http://www.osfree.org/doku/> - **osFree wiki**

Permanent link:
http://www.osfree.org/doku/doku.php?id=en:ibm:ifs:mechanism:init:extboot

Last update: **2014/05/12 01:47**