

-- FakeSegDefs.mesa Edited by: Sandman, February 3, 1978 10:02 AM

## DIRECTORY

SegmentDefs: FROM "segmentdefs",  
AltoDefs: FROM "altodefs",  
WartDefs: FROM "wartdefs";

DEFINITIONS FROM SegmentDefs, AltoDefs;

FakeSegDefs: DEFINITIONS =

BEGIN

DefaultBase: PageNumber = SegmentDefs.DefaultBase;  
DefaultPages: PageCount = SegmentDefs.DefaultPages;  
DefaultFile: FileHandle = NIL;

VMBounds: TYPE = RECORD [ffvmp, lfvmp: CARDINAL];

FakeSegmentHandle: TYPE = POINTER TO FakeSegmentObject;

FakeSegmentObject: TYPE = RECORD [  
File: FileHandle,  
Type: PRIVATE {free, file, window},  
Class: FileSegmentClass,  
Access: AccessOptions,  
SwappedIn: BOOLEAN,  
CopyToImage: BOOLEAN,  
Base: PageNumber,  
Pages: PageCount,  
VMaddress: Address,  
Link: FakeSegmentHandle,  
LockCount: PRIVATE INTEGER,  
TrueFile: FileHandle,  
ImageBase: PageNumber,  
Link2: FileSegmentHandle,  
BootLink: PRIVATE POINTER];

FakeInvalidSegment: ERROR [s:POINTER];

FakeNewSegment: PROCEDURE [file:FileHandle, base:PageNumber, pages:PageCount, access:AccessOptions] R  
\*\*ETURNS [FakeSegmentHandle];

FakeDeleteSegment: PROCEDURE [segment:FakeSegmentHandle];

FakeTooManySegments: SIGNAL;  
FakeInvalidSegmentSize: ERROR;  
FakeInsufficientVM: SIGNAL [needed:PageCount];  
FakeVMNotFree: SIGNAL [base:PageNumber, pages:PageCount];

FakeSwapError: ERROR [segment:FakeSegmentHandle];  
FakeSwapIn, FakeUnlock, FakeSwapOut: PROCEDURE [segment:FakeSegmentHandle];

FakeEnumerateSegments: PROCEDURE [  
proc:PROCEDURE[FakeSegmentHandle]RETURNS[BOOLEAN]] RETURNS [FakeSegmentHandle];

FakeFileHandle: TYPE = POINTER TO FakeFileObject;

FakeFileObject: TYPE = RECORD [  
File: FileHandle,  
InUse: PRIVATE BOOLEAN,  
ShouldOpen: BOOLEAN,  
Access: AccessOptions,  
BootLink: PRIVATE POINTER];

FakeTooManyFiles: SIGNAL;

FakeInitFileMachinery: PROCEDURE;  
FakeInsertFile: PROCEDURE [seg:FakeSegmentHandle] RETURNS [BOOLEAN];  
FakeInitSegMachinery: PROCEDURE [PageNumber, PageNumber];  
FakeEnumerateFiles: PROCEDURE [  
proc:PROCEDURE[FakeFileHandle]RETURNS[BOOLEAN]] RETURNS [FakeFileHandle];  
DeclareMemoryBounds: PROCEDURE [CARDINAL, CARDINAL];  
GetVMBounds: PROCEDURE RETURNS [VMBounds];  
SetSegmentBootLink: PROCEDURE [FakeSegmentHandle, WartDefs.BootIndex];  
GetSegmentBootLink: PROCEDURE [FakeSegmentHandle] RETURNS [WartDefs.SegmentBootIndex];  
END.