# FS_ALLOCATEPAGESPACE

## Purpose

Changes the size the paging file on disk.

## Calling Sequence

```
int far pascal FS_ALLOCATEPAGESPACE(psffsi, psffsd, ulsize, ulWantContig)

struct sffsi far * psffsi;
struct sffsd far * psffsd;
unsigned long ulsize;
unsigned short ulWantContig;
```

## Where

*psffsi* is a pointer to the file-system-independent portion of an open file instance.

*psffsd* is a pointer to the file-system-dependent portion of an open file instance.

*ulsize* is the desired new size of the paging file. If the new size is smaller than the current size, the excess space is released. If the new size is larger than the current size, the requested size is allocated.

*ulWantContig* indicates the minimum contiguity requirement (in bytes).

## Remarks

*ulWantContig* is a demand for contiguity. If *ulWantContig* is non-zero(0), the FSD must allocate any space in the swap file that is not contiguous in *ulWantContig* chunks on *ulWantContig* boundaries. If it is not possible to grow the file to ulSize bytes meeting the *ulWantContig* requirement, the operation should fail. If the file is being shrunk *ulWantContig* is irrelevant and should be ignored.

FSDs that support the paging I/O interface should be expected to be sensible in allocating page space. In particular, they are expected to always attempt to allocate space such that *ulWantContig* sized blocks on *ulWantContig* boundaries are physically contiguous on disk, and to keep the page file as a whole contiguous as possible.